

A Wavelet Based Algorithm for
Image Super-Resolution

By

Zhen Xie

B.S., University of Science and Technology of China, 2003

Adviser: James G. Nagy, Ph.D.

A thesis submitted to the Faculty of the Graduate
School of Emory University in partial fulfillment of
the requirements for the degree of
Master of Science

Department of Math and Computer Science

2007

Table of Contents

1	Introduction to Image Super-Resolution	1
2	Introduction to Wavelet Transforms	6
2.1	Haar Wavelet Transform	7
2.1.1	1D Case	7
2.1.2	2D Case	21
3	Wavelet Based Image Super-Resolution	25
3.1	Previous Work	25
3.1.1	Frequency Domain Methods	25
3.1.2	Spatial Domain Methods	26
3.2	Intuitive Algorithm	29
3.2.1	1D Curve	29
3.2.2	2D Images	32
3.3	Matrix Form Algorithm	35
3.3.1	1D and Level-1 Up Super-Resolution	36
3.3.2	Relation Between the Intuitive and Matrix Form Algorithm	40
3.3.3	Landweber Iteration	46
3.3.4	1D and Multilevel up Super-Resolution	48
3.3.5	One Step Solution in the 1D Case	50
3.3.6	2D and Level-1 Up Super-Resolution	54
3.3.7	2D and Multilevel Up Super-Resolution	55
4	Summary	56

TABLE OF CONTENTS

References	57
------------	----

List of Figures

1	Illustration of the same image at different pixel resolution.	1
2	Illustration of sub-pixel shift and LR image superposition. (a) represents the original image with 16×16 pixels, (b) is the level-1 down low resolution image of (a); every pixel in (b) is the average of four pixels in (a). After shifting all the pixels in (a) one pixel right, and averaging four pixel values in the shifted (a), then (c) is the new level-1 down low resolution image. (d) and (e) are similar to (c), and the difference is the shift directions. The subscripts for (b), (c), (d) and (e) indicate the shifting direction of the original image. (f) is the superposition of all the low resolution images (b), (c), (d) and (e) according to their relative position.	4
3	The box basis for V^2	11
4	The Haar wavelets for W^2	12
5	Schematic illustration of 1D wavelet transform. The shaded area represents the “trend”.	21

LIST OF FIGURES

6	<p>Schematic illustration of a 2D wavelet transform. The shaded area represents the “trend”. (a) is the original matrix. After applying the wavelet transform on each row of the original matrix, (b) is obtained. Applying the wavelet transform on each column in the matrix as shown in (b), (c) is then generated. (c) is the final matrix of the wavelet transform of the matrix in (a). Continue the wavelet transform on the “row” and “column” of the trend in (c), then (d) and (e) are obtained in order. The overall process from (a) to (c) is shown as (f) to (g). The process from (c) to (e) is shown as (g) to (h). The wavelet transform on a matrix can continue to a third level (from (h) to (i)) and fourth level (from (i) to (j)) and even more levels.</p>	22
7	<p>Schematic illustration of 2D wavelet transforms on an image. (a) is the original image, (b) is the original level-1 wavelet transformed image in (a), (c) is an amplitude adjustment of (b) to match the same brightness of (a), and (d) is the level-2 wavelet transform of the image in (a) with adjusted amplitude.</p>	23
8	<p>Illustration of intuitive wavelet based super-resolution algorithm applied to a 1D example (left) and a 2D example (right).</p>	31
9	<p>1D level-1 up super-resolution error convergence.</p>	32
10	<p>2D level-1 up super-resolution. (a) is the original high resolution image, (b) is one of the four low resolution images generated from (a), and (c) is the super-resolution image after about 110 iterations.</p>	33
11	<p>2D level-1 up super-resolution error convergence.</p>	34

LIST OF FIGURES

12	2D level-2 up super-resolution. (a) is the original high resolution image, (b) is one of the 16 low resolution images generated from (a), and (c) is the super-resolution image after about 600 iterations.	34
13	2D level-2 up super-resolution error convergence.	35
14	Decomposition of 1D level-2 up super-resolution problem into two level- 1 up process.	54

List of Tables

1	Test of the intuitive algorithm on a simple 1D curve.	30
---	---	----

1 Introduction to Image Super-Resolution

Images with high resolution (HR) are always desired and often required, in most electronic imaging applications. HR means that pixel density within an image is high, and therefore an HR image can offer more details. Fig. 1 is an illustration of how the same image might appear at different pixel resolutions, if the pixels were poorly rendered as sharp squares (normally, a smooth image reconstruction from pixels would be preferred, but for illustration of pixels, the sharp squares make the point better).



Figure 1: Illustration of the same image at different pixel resolution.

The more details that can be provided by an HR image may be critical in various applications. For example, it is very helpful for a doctor to make a correct diagnosis based on HR medical images. In astronomy, an object can be easily distinguished from similar ones using HR satellite images. In computer vision, the performance of pattern recognition can be improved if an HR image is provided. As is known, since the 1970s digital images are typically captured by a charge-coupled device (CCD) or by a CMOS image sensor. Although these sensors are suitable for most imaging applications, the current resolution level and consumer price will not satisfy the future demand. For example, people want an inexpensive HR digital camera/camcorder, and scientists often need a very HR level close to that of an analog 35 mm film that has

no visible artifacts when an image is magnified. As a result, approaches to increase the current resolution level are needed.

The most direct solution to increase spatial resolution is to reduce the pixel size (i.e., increase the number of pixels per unit area) by sensor manufacturing techniques. As the pixel size decreases, however, the amount of light that can be captured by each pixel also decreases. This generates something known as shot noise that severely degrades image quality. Thus, there are limitations of the pixel size reduction without suffering effects of shot noise.

The current image sensor technology has almost reached this level. Another approach for enhancing the spatial resolution is to increase the chip size, which leads to an increase in capacitance [Komatsu 1993]. Since large capacitance makes it difficult to speed up a charge transfer rate, this approach is not considered effective. The high cost for high precision optics and image sensors is also an important concern in many commercial applications regarding HR imaging. Therefore, a new approach toward increasing spatial resolution is required to overcome these limitations of the sensors and optics manufacturing technology. One promising approach is to use signal processing techniques to obtain an HR image (or sequence) from observed multiple low-resolution (LR) images. Recently, such a resolution enhancement approach has been one of the most active research areas, and it is called super-resolution (SR) image reconstruction or simply resolution enhancement in the literature [Park 2003].

The major advantages of the signal processing approach is that it may cost less and the existing LR imaging systems can be still utilized. SR image reconstruction has proven to be useful in many practical cases where multiple frames of the same scene can be obtained, including medical imaging, satellite imaging, and video ap-

plications. One application is to reconstruct a higher quality digital image from LR images obtained with an inexpensive LR camera/camcorder for printing or frame freeze purposes. The SR technique is also useful in medical imaging such as computed tomography (CT) and magnetic resonance imaging (MRI) since the acquisition of multiple images is possible while the resolution quality is limited. In satellite imaging applications such as remote sensing, several images of the same area are usually provided, and the SR technique to improve the resolution of the target can be considered.

How can we obtain an HR image from multiple LR images? The basic premise for increasing the spatial resolution in SR techniques is the availability of multiple LR images that capture different information from the same scene. In SR, typically, the LR images represent different “looks” at the same scene. That is, LR images are sub-sampled (aliased) as well as shifted with subpixel precision. If the LR images are shifted by integer pixel units, then each image contains the same information, and thus there is no new information that can be used to reconstruct an HR image. If the LR images have different subpixel shifts from each other and if aliasing is present, however, then each image cannot be obtained from the others. In this case, the new information contained in each LR image can be exploited to obtain an HR image. To obtain different looks at the same scene, some relative scene motions must exist from frame to frame via multiple scenes or video sequences. Multiple scenes can be obtained from one camera with several captures or from multiple cameras located in different positions. These scene motions can occur due to the controlled motions in imaging systems, e.g., images acquired from orbiting satellites. The same is true of uncontrolled motions, e.g., movement of local objects or vibrating imaging systems. If

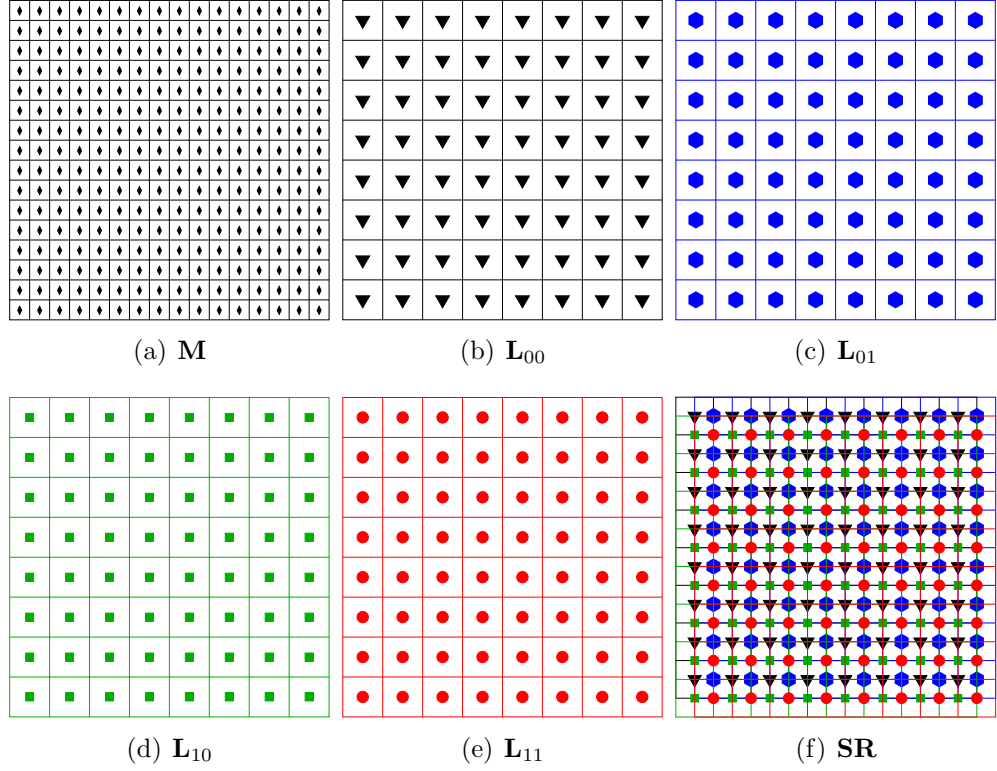


Figure 2: Illustration of sub-pixel shift and LR image superposition. (a) represents the original image with 16×16 pixels, (b) is the level-1 down low resolution image of (a); every pixel in (b) is the average of four pixels in (a). After shifting all the pixels in (a) one pixel right, and averaging four pixel values in the shifted (a), then (c) is the new level-1 down low resolution image. (d) and (e) are similar to (c), and the difference is the shift directions. The subscripts for (b), (c), (d) and (e) indicate the shifting direction of the original image. (f) is the superposition of all the low resolution images (b), (c), (d) and (e) according to their relative position.

these scene motions are known or can be estimated within subpixel accuracy and if we combine these LR images, SR image reconstruction is possible as illustrated in Fig 2. In the process of recording a digital image, there is a natural loss of spatial resolution caused by the optical distortions (defocus, diffraction limit, etc.), motion blur due to limited shutter speed, noise that occurs within the sensor or during transmission, and insufficient sensor density. Thus, the recorded image usually suffers from blur, noise, and aliasing effects. Although some SR algorithms cover image restoration techniques that produce high quality images from noisy, blurred images, the main concern of an SR algorithm is to reconstruct HR images from undersampled LR images. Therefore, the goal of SR techniques is to restore an HR image from several degraded and aliased LR images which is also the main goal of this thesis.

In the following section, we present a brief introduction to wavelet transforms, with particular focus on Haar wavelet transforms. The various algorithms developed based on wavelet transforms are discussed in the third section and a summary of the algorithms developed in this thesis is given in the final section.

2 Introduction to Wavelet Transforms

Wavelets are a mathematical tool for hierarchically decomposing functions. They allow a function to be described in terms of a coarse overall shape, plus details that range from broad to narrow. Regardless of whether the function of interest is an image, a curve, or a surface, wavelets offer an elegant technique for representing the levels of detail in the function.

Although wavelets have their roots in approximation theory [Daubechies 1988] and signal processing [Mallet 1989], they have recently been applied to many problems in computer graphics. These graphics applications include image editing [Berman 1994], image compression [DeVore 1992] and image querying [Jacobs 1995]. There are also some other applications in automatic level-of-detail control for editing and rendering curves and surfaces [Finkelstein 1994, Gortler and Cohen 1995], surface reconstruction from contours [Meyers 1994], fast methods for solving simulation problems in animation [Liu 1994] and global illumination [Christensen 1995, Gortler 1995, Schröder 1994].

In this thesis, we want to apply wavelet transforms to the image super-resolution problem. Before showing the intrinsic relation between super-resolution and wavelet transforms, we need a brief understanding of wavelet transforms. Since only the “Haar” wavelet is used in the algorithm discussed in this thesis, we restrict our introduction to the Haar wavelet transform.

The following two subsections are devoted to one-dimensional and two-dimensional wavelet transforms. The wavelet basis function and matrix representation are also presented.

2.1 Haar Wavelet Transform

The Haar basis has an intrinsic relation with super-resolution, and hence will be discussed here. In the first following section, we will explain the one-dimensional wavelet transform in detail, and then follow up with the two-dimensional case.

2.1.1 1D Case

To get a sense for how wavelets work, let's start with a simple example. Suppose we are given a one-dimensional “image” with a resolution of four pixels, having values

$$(5 \ 7 \ 3 \ 1)$$

We can represent this image in the Haar basis by computing a wavelet transform. To do this, we first average the pixels together, pairwise, to get the low resolution image with pixel values

$$(6 \ 2) .$$

$(6 \ 2)$ may be a good approximation to $(5 \ 7 \ 3 \ 1)$, but clearly, some information has been lost in this averaging process. To recover the original four pixel values from the two averaged values, we need to store some detail coefficients that capture the missing information. In this example, we choose -1 for the first detail coefficient, since the average we computed is 1 more than 5 and 1 less than 7. This single number allows us to recover the first two pixels of our original four-pixel image. Similarly, the second detail coefficient is 1, since $2 + 1 = 3$ and $2 - 1 = 1$. Thus, we have decomposed the original image into a lower resolution (two-pixel) version and a pair of detail coefficients. Repeating this process recursively on the averages gives

2.1 Haar Wavelet Transform

the full decomposition:

Level	Resolution	Averages	Detail coefficients	Result
0	4	(5 7 3 1)		(5 7 3 1)
1	2	(6 2)	(-1 1)	(6 2 -1 1)
2	1	(4)	(2)	(4 2 -1 1)

Here we notice that the averaging operation is performed twice (one is from resolution 4 to 2, and the other is from resolution 2 to 1), and hence the whole operation described above is referred to as a level-2 transformation. The “average” in each step is called the “approximation” or “trend” without distinction and the “detail coefficients” are called “detail”.

Finally, we define the wavelet transform (also called the wavelet decomposition) of the original four-pixel image to be the single coefficient representing the overall average of the original image, followed by the detail coefficients in order of increasing resolution. Thus, for the one-dimensional Haar basis, the wavelet transform of our original four-pixel image is given by

$$(4 \ 2 \ -1 \ 1) .$$

Note that no information has been gained or lost by this process. The original image had four coefficients, and so does the transform. Also note that, given the transform, we can reconstruct the image to any resolution by recursively adding and subtracting the detail coefficients from the lower resolution versions. Storing the image’s wavelet transform, rather than the image itself, has a number of advantages. One advantage of the wavelet transform is that often a large number of the detail coefficients turn out to be very small in magnitude. Truncating, or removing, these

2.1 Haar Wavelet Transform

small coefficients from the representation introduces only small errors in the reconstructed image, giving a form of “lossy” image compression. The lower resolution image is obtained by discarding all the detail coefficients from some level, and this is obviously closely related to super-resolution.

To make the above example look more mathematical and general, suppose f is a row vector of size 2^n , and apply the Haar wavelet transform on it once to get the trend a^1 and detail d^1 as

$$f \xrightarrow{\text{HWT}} (a^1 | d^1). \quad (2.1)$$

where a^1 and d^1 are vectors of size 2^{n-1} ; a^1 is the level-1 “trend” and d^1 is the level-1 detail.

If we continue performing the wavelet transform on a^1 , we can get

$$a^1 \xrightarrow{\text{HWT}} (a^2 | d^2), \quad (2.2)$$

and we can continue the above process until the trend consists of only one element.

The overall process may be depicted as follows:

$$f \xrightarrow{\text{level-1}} (a^1 | d^1) \xrightarrow{\text{level-2}} (a^2 | d^2 | d^1) \rightarrow \dots \xrightarrow{\text{level-}j} (a^j | d^j | d^{j-1} | \dots | d^1). \quad (2.3)$$

Clearly, an analogous notation can be used if f is a column vector rather than a row vector.

We have shown how one-dimensional images can be treated as sequences of coefficients. Alternatively, we can think of images as piecewise-constant functions on the half-open interval $[0, 1)$. To do so, we will use the concept of a vector space from linear algebra. A one-pixel image is just a function that is constant over the entire

2.1 Haar Wavelet Transform

interval $[0, 1)$. We'll let V^0 be the vector space of all these functions. A two-pixel image has two constant pieces over the intervals $[0, \frac{1}{2})$ and $[\frac{1}{2}, 1)$. We'll call the space containing all these functions V^1 . If we continue in this manner, the space V^j will include all piecewise-constant functions defined on the interval $[0, 1)$ with constant pieces over each of 2^j equal length subintervals.

We can now think of every one-dimensional image with 2^j pixels as an element, or vector, in V^j . Note that because these vectors are all functions defined on the unit interval, every vector in V^j is also contained in V^{j+1} . For example, we can always describe a piecewise-constant function with two intervals as a piecewise-constant function with four intervals, with each interval in the first function corresponding to a pair of intervals in the second. Thus, the spaces V^j are nested; that is,

$$V^0 \subset V^1 \subset V^2 \subset \dots$$

Now we need to define a basis for each vector space V^j . The basis functions for the spaces V^j are called *scaling functions*, and are usually denoted by the symbol ϕ . A simple basis for V^j is given by the set of scaled and translated “box” functions:

$$\phi_i^j(x) := \phi(2^j x - i), \quad i = 0, \dots, 2^j - 1.$$

where

$$\phi(x) := \begin{cases} 1 & \text{for } 0 \leq x < 1 \\ 0 & \text{otherwise.} \end{cases}$$

As an example, Fig. 3 shows the four box functions forming a basis for V^2 .

The next step is to choose an inner product defined on the vector space V^j .

2.1 Haar Wavelet Transform

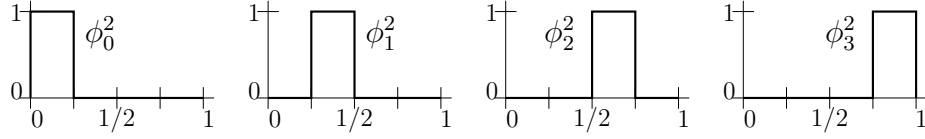


Figure 3: The box basis for V^2 .

Consider the “standard” inner product,

$$\langle f|g \rangle := \int_0^1 f(x)g(x) dx,$$

for two elements $f, g \in V^j$. We can then define a new vector space W^j as the *orthogonal complement* of V^j in V^{j+1} . In other words, we will let W^j be the space of all functions in V^{j+1} that are orthogonal to all functions in V^j under the chosen inner product. Informally, we can think of the wavelets in W^j as a means for representing the parts of a function in V^{j+1} that cannot be represented in V^j .

A collection of linearly independent functions $\psi_i^j(x)$ spanning W^j is called a *wavelet*. These basis functions have two important properties.

1. The basis function ψ_i^j of W^j , together with the basis function ϕ_i^j of V^j , form a basis for V^{j+1} .
2. Every basis function ψ_i^j of W^j is orthogonal to every basis function ϕ_i^j of V^j under the chosen inner product.

The wavelets corresponding to the box basis are known as the *Haar* wavelets, given by

$$\psi_i^j := \psi(2^j x - i), \quad i = 0, \dots, 2^j - 1,$$

2.1 Haar Wavelet Transform

where

$$\psi(x) := \begin{cases} 1 & \text{for } 0 \leq x < 1/2 \\ -1 & \text{for } 1/2 \leq x < 1 \\ 0 & \text{otherwise.} \end{cases}$$

Fig. 4 shows the four Haar wavelets spanning W^2 .

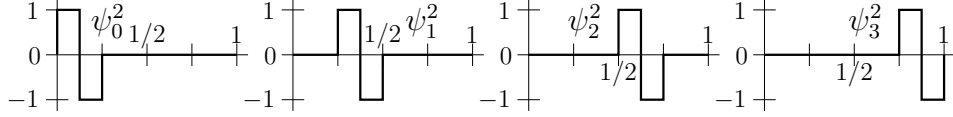


Figure 4: The Haar wavelets for W^2 .

If we represent the original image $\begin{pmatrix} 5 & 7 & 3 & 1 \end{pmatrix}$ as a piecewise-constant function $\mathcal{I}(x)$ on $[0, 1)$, then we can express $\mathcal{I}(x)$ as a linear combination of the box functions in V^2 :

$$\mathcal{I}(x) = c_0^2 \phi_0^2(x) + c_1^2 \phi_1^2(x) + c_2^2 \phi_2^2(x) + c_3^2 \phi_3^2(x).$$

Note that the coefficients c_0^2, \dots, c_3^2 are just the four original pixel values $\begin{pmatrix} 5 & 7 & 3 & 1 \end{pmatrix}$.

Since

$$V^2 = V^1 \oplus W^1,$$

we can rewrite the expression for $\mathcal{I}(x)$ in terms of the basis functions in V^1 and W^1 , using pairwise averaging and differencing:

$$\mathcal{I}(x) = c_0^1 \phi_0^1(x) + c_1^1 \phi_1^1(x) + d_0^1 \psi_0^1(x) + d_1^1 \psi_1^1(x).$$

These four coefficients are $\begin{pmatrix} 6 & 2 & -1 & 1 \end{pmatrix}$ which are quite familiar. Since $V^1 =$

2.1 Haar Wavelet Transform

$V^0 \oplus W^0$, we can further represent

$$\mathcal{I}(x) = c_0^0 \phi_0^0(x) + d_0^0 \psi_0^0(x) + d_0^1 \psi_0^1(x) + d_1^1 \psi_1^1(x),$$

and the four coefficients are just $\begin{pmatrix} 4 & 2 & -1 & 1 \end{pmatrix}$ which is the level-2 wavelet transform of $\begin{pmatrix} 5 & 7 & 3 & 1 \end{pmatrix}$. Instead of using the four usual box functions, we can use $\phi_0^0(x)$, $\psi_0^0(x)$, $\psi_0^1(x)$ and $\psi_1^1(x)$ to represent the overall average, the broad detail, and the two types of finer detail possible for a function in V^2 . The Haar basis for V^j with $j > 2$ includes these functions as well as narrower translates of the wavelet $\psi(x)$.

In order to represent the above wavelet transform in matrix form, we define the level-1 *wavelets* as

$$\begin{aligned} w_1^1 &= \left(\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0, 0, \dots, 0 \right) \\ w_2^1 &= \left(0, 0, \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}, 0, 0, \dots, 0 \right) \\ &\vdots \\ w_{n/2}^1 &= \left(0, 0, \dots, 0, \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}} \right), \end{aligned} \tag{2.4}$$

and the *scaling signals* as

$$\begin{aligned} v_1^1 &= \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, 0, \dots, 0 \right) \\ v_2^1 &= \left(0, 0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0, 0, \dots, 0 \right) \\ &\vdots \\ v_{n/2}^1 &= \left(0, 0, \dots, 0, \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}} \right). \end{aligned} \tag{2.5}$$

2.1 Haar Wavelet Transform

Note that the non-zero numbers for w_i^1 and v_i^1 are $\pm \frac{1}{\sqrt{2}}$, this is to ensure that every vector has unit length.

Now stack all the level-1 wavelets together to construct an $\frac{n}{2} \times n$ matrix \mathbf{W} as

$$\mathbf{W} = \begin{pmatrix} w_1^1 \\ w_2^1 \\ \vdots \\ w_{n/2}^1 \end{pmatrix} \quad (2.6)$$

and all the level-1 scaling signals to construct an $\frac{n}{2} \times n$ matrix \mathbf{V} as

$$\mathbf{V} = \begin{pmatrix} v_1^1 \\ v_2^1 \\ \vdots \\ v_{n/2}^1 \end{pmatrix}. \quad (2.7)$$

Definition The matrix constructed as in Eq. 2.6 is called a **Wavelet Matrix**.

Definition The matrix constructed as in Eq. 2.7 is called a **Scaling Matrix**.

Definition In space $\Re^{n \times n}$, the level-1 scaling matrix \mathbf{V}_1 is defined as in Eq. 2.7 of size $\frac{n}{2} \times n$, the level-2 scaling matrix \mathbf{V}_2 is the upper left corner of \mathbf{V}_1 of the size $\frac{n}{4} \times \frac{n}{2}$. Generally, the level- k scaling matrix \mathbf{V}_k is the upper left corner of \mathbf{V}_1 of size $\frac{n}{2^k} \times \frac{n}{2^{k-1}}$. Similarly, the level- k wavelet matrix \mathbf{W}_k is the upper left corner of the level-1 wavelet matrix \mathbf{W}_1 as defined in Eq. 2.6 with the same dimension as \mathbf{V}_k .

Using the *wavelet matrix* and *scaling matrix*, the level-1 Haar wavelet transform

2.1 Haar Wavelet Transform

of a column vector f can be written in matrix form as:

$$a^1 = \mathbf{V}_1 \cdot f$$

$$d^1 = \mathbf{W}_1 \cdot f$$

or more compactly as

$$\begin{pmatrix} a^1 \\ d^1 \end{pmatrix} = \begin{pmatrix} \mathbf{V}_1 \\ \mathbf{W}_1 \end{pmatrix} \cdot f$$

Similarly, if f is a row vector, $a^1 = f \cdot \mathbf{V}_1^T$ and $d^1 = f \cdot \mathbf{W}_1^T$, or more compactly as $(a^1, d^1) = f \cdot (\mathbf{V}_1^T, \mathbf{W}_1^T)$.

Theorem 2.1 *The level-1 Haar wavelet transform of a column vector $f \in \mathfrak{R}^n$ can be written as*

$$HWT(f) = \begin{pmatrix} a \\ d \end{pmatrix} = \begin{pmatrix} \mathbf{V} \\ \mathbf{W} \end{pmatrix} \cdot f.$$

where \mathbf{V} and \mathbf{W} are the scaling and wavelet matrices as defined in Eq. 2.7 and Eq. 2.6, respectively. If f is a row vector, then

$$HWT(f) = (a, d) = f \cdot (\mathbf{V}^T, \mathbf{W}^T).$$

Theorem 2.2 *If $\mathbf{V}, \mathbf{W} \in \mathfrak{R}^{\frac{n}{2} \times n}$, and \mathbf{V} is the scaling matrix, and \mathbf{W} is the wavelet matrix, then $\mathbf{V}^T \mathbf{V} + \mathbf{W}^T \mathbf{W} = \mathbf{I}$*

Proof Since

$$\mathbf{W} = \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_{n/2} \end{pmatrix}$$

and

$$\mathbf{V} = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_{n/2} \end{pmatrix}$$

according to Eq. 2.6 and Eq. 2.7 (note that we dropped the superscripts of the w and v vectors to simplify the notation), it follows that

$$\mathbf{W}^T = \begin{pmatrix} w_1^T & w_2^T & \cdots & w_{n/2}^T \end{pmatrix} \quad (2.8)$$

and

$$\mathbf{V}^T = \begin{pmatrix} v_1^T & v_2^T & \cdots & v_{n/2}^T \end{pmatrix}. \quad (2.9)$$

It follows that,

$$\begin{aligned}
& \mathbf{V}^T \mathbf{V} + \mathbf{W}^T \mathbf{W} \\
&= \begin{pmatrix} v_1^T & \cdots & v_i^T & \cdots & v_{n/2}^T \end{pmatrix} \cdot \begin{pmatrix} v_1 \\ \vdots \\ v_i \\ \vdots \\ v_{n/2} \end{pmatrix} + \\
&\quad \begin{pmatrix} w_1^T & \cdots & w_i^T & \cdots & w_{n/2}^T \end{pmatrix} \cdot \begin{pmatrix} w_1 \\ \vdots \\ w_i \\ \vdots \\ w_{n/2} \end{pmatrix} \tag{2.10} \\
&= v_1^T v_1 + \cdots + v_i^T v_i + \cdots + v_{n/2}^T v_{n/2} + w_1^T w_1 + \cdots + w_i^T w_i + \cdots + w_{n/2}^T w_{n/2} \\
&= (v_1^T v_1 + w_1^T w_1) + \cdots + (v_i^T v_i + w_i^T w_i) + \cdots + (v_{n/2}^T v_{n/2} + w_{n/2}^T w_{n/2}).
\end{aligned}$$

2.1 Haar Wavelet Transform

Because

$$v_i^T v_i = \begin{matrix} & 1 & \cdots & 2i-2 & 2i-1 & 2i & 2i+1 & \cdots & n \\ \begin{matrix} 1 \\ \vdots \\ 2i-2 \\ 2i-1 \\ 2i \\ 2i+1 \\ 2i+2 \\ \vdots \\ n \end{matrix} & \begin{pmatrix} 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & \frac{1}{2} & \frac{1}{2} & 0 & \cdots & 0 \\ 0 & \cdots & 0 & \frac{2}{2} & \frac{1}{2} & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \end{pmatrix} \end{matrix} \quad (2.11)$$

and

$$w_i^T w_i = \begin{matrix} & 1 & \cdots & 2i-2 & 2i-1 & 2i & 2i+1 & \cdots & n \\ \begin{matrix} 1 \\ \vdots \\ 2i-2 \\ 2i-1 \\ 2i \\ 2i+1 \\ 2i+2 \\ \vdots \\ n \end{matrix} & \begin{pmatrix} 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & \frac{1}{2} & -\frac{1}{2} & 0 & \cdots & 0 \\ 0 & \cdots & 0 & -\frac{2}{2} & \frac{1}{2} & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \end{pmatrix}, \end{matrix} \quad (2.12)$$

2.1 Haar Wavelet Transform

we obtain,

$$v_i^T v_i + w_i^T w_i = \begin{matrix} & 1 & \cdots & 2i-2 & 2i-1 & 2i & 2i+1 & \cdots & n \\ \begin{matrix} 1 \\ \vdots \\ 2i-2 \\ 2i-1 \\ 2i \\ 2i+1 \\ 2i+2 \\ \vdots \\ n \end{matrix} & \begin{pmatrix} 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 1 & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & 1 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 \end{pmatrix} \end{matrix}. \quad (2.13)$$

Finally,

$$\mathbf{V}^T \mathbf{V} + \mathbf{W}^T \mathbf{W} = (v_1^T v_1 + w_1^T w_1) + \cdots + (v_i^T v_i + w_i^T w_i) + \cdots + (v_{n/2}^T v_{n/2} + w_{n/2}^T w_{n/2}) = \mathbf{I}. \quad \blacksquare \quad (2.14)$$

Definition The level-1 inverse Haar wavelet transform of a column vector

$$\begin{pmatrix} a \\ d \end{pmatrix}$$

is

$$IHW T \begin{pmatrix} a \\ d \end{pmatrix} = (\mathbf{V}^T, \mathbf{W}^T) \begin{pmatrix} a \\ d \end{pmatrix} = \mathbf{V}^T \cdot a + \mathbf{W}^T \cdot d = f.$$

2.1 Haar Wavelet Transform

where \mathbf{V} and \mathbf{W} are the scaling and wavelet matrices as defined in Eq. 2.7 and Eq. 2.6, respectively.

The level-1 inverse Haar wavelet transform of a row vector (a, d) is

$$\text{IHWIT}(a, d) = (a, d) \cdot \begin{pmatrix} \mathbf{V} \\ \mathbf{W} \end{pmatrix} = a\mathbf{V} + d\mathbf{W} = f.$$

This definition of *inverse Haar wavelet transform* is an immediate result of the Theorem 2.2.

Theorem 2.3 *Let $\mathbf{V}_k, \mathbf{W}_k \in \mathfrak{R}_{2^k \times \frac{n}{2^{k-1}}}$ be the level- k scaling and wavelet matrix in $\mathfrak{R}^{n \times n}$, then the level- k Haar wavelet transform of $f \in \mathfrak{R}^n$ is performed as*

$$f \xrightarrow{\text{level-1 HWT}} \begin{pmatrix} \mathbf{V}_1 \\ \mathbf{W}_1 \end{pmatrix} f \xrightarrow{\text{level-2 HWT}} \begin{pmatrix} \mathbf{V}_2 \mathbf{V}_1 \\ \mathbf{W}_2 \mathbf{V}_1 \\ \mathbf{W}_1 \end{pmatrix} f \rightarrow \dots \xrightarrow{\text{level-}k \text{ HWT}} \begin{pmatrix} \mathbf{V}_k \mathbf{V}_{k-1} \cdots \mathbf{V}_1 \\ \mathbf{W}_k \mathbf{V}_{k-1} \cdots \mathbf{V}_1 \\ \vdots \\ \mathbf{W}_3 \mathbf{V}_2 \mathbf{V}_1 \\ \mathbf{W}_2 \mathbf{V}_1 \\ \mathbf{W}_1 \end{pmatrix} f$$

A schematic illustration of the 1D wavelet transform can be shown in Fig. 5. The wavelet transform only acts on the “trend” which is shown as the shaded region in Fig. 5. The “trend” is always cut into two pieces, and this process can continue until

2.1 Haar Wavelet Transform

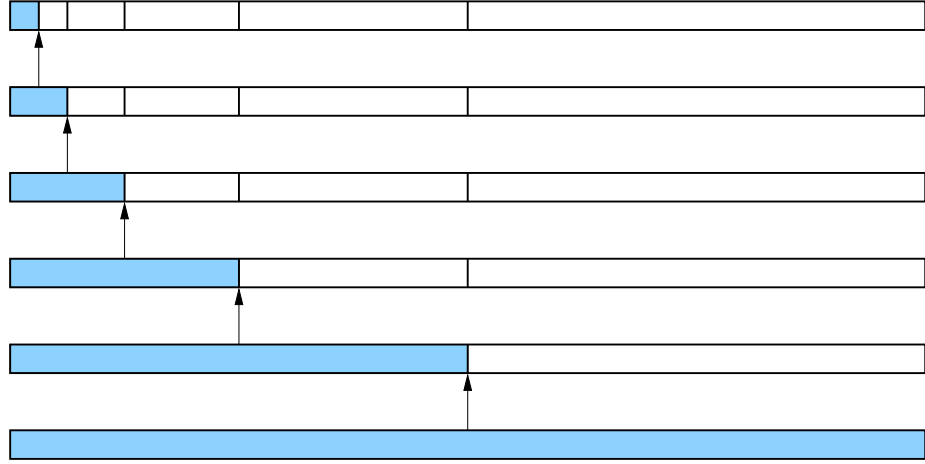


Figure 5: Schematic illustration of 1D wavelet transform. The shaded area represents the “trend”.

there is only one element left in the “trend”.

The inverse multilevel Haar wavelet transform is just the inverse procedure of the multilevel forward wavelet transform. It combines the “trend” and detail at the same level to get the “trend” for the lower level. Continue this process until all the “trend” and “detail” components are combined.

2.1.2 2D Case

In the 2D case, it is a little different since now we have a square matrix instead of a vector. There are basically two approaches to do wavelet transforms on a square matrix. The first approach is to do wavelet transforms on each row first, and then do wavelet transforms on each of the resulting columns. As a result, the “trend” stays in the upper-left corner of the matrix. Another approach is to alternate between rows and columns. That is, begin with a wavelet transform on the first row, and then do a wavelet transform on the first column, then continue with wavelet transforms on the second row and the second column, so on and so forth. Finally, the “trend” still lives

2.1 Haar Wavelet Transform

in the upper left corner of the matrix, and all the other elements are the “detail” of the matrix.

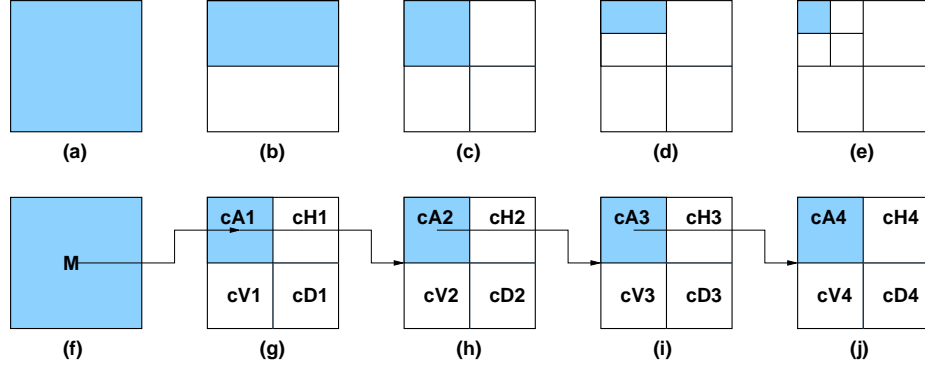


Figure 6: Schematic illustration of a 2D wavelet transform. The shaded area represents the “trend”. (a) is the original matrix. After applying the wavelet transform on each row of the original matrix, (b) is obtained. Applying the wavelet transform on each column in the matrix as shown in (b), (c) is then generated. (c) is the final matrix of the wavelet transform of the matrix in (a). Continue the wavelet transform on the “row” and “column” of the trend in (c), then (d) and (e) are obtained in order. The overall process from (a) to (c) is shown as (f) to (g). The process from (c) to (e) is shown as (g) to (h). The wavelet transform on a matrix can continue to a third level (from (h) to (i)) and fourth level (from (i) to (j)) and even more levels.

A schematic illustration of the 2D wavelet transform can be shown in Fig. 6 and its application on a real 2D image is shown in Fig. 7.

In matrix representation, let \mathbf{V} and \mathbf{W} be the scaling and wavelet matrix, then the level-1 Haar wavelet transform of a square matrix \mathbf{M} can be written as:

$$\mathbf{M} \xrightarrow{\text{HWT}} \begin{pmatrix} \mathbf{V} \\ \mathbf{W} \end{pmatrix} \mathbf{M} (\mathbf{V}^T, \mathbf{W}^T) = \begin{pmatrix} \mathbf{VMV}^T & \mathbf{VMW}^T \\ \mathbf{WMV}^T & \mathbf{WMW}^T \end{pmatrix}. \quad (2.15)$$

In MATLAB, the output of `dwt2(M)` is four small matrices, $[\mathbf{cA}, \mathbf{cH}, \mathbf{cV}, \mathbf{cD}]$. In the

2.1 Haar Wavelet Transform

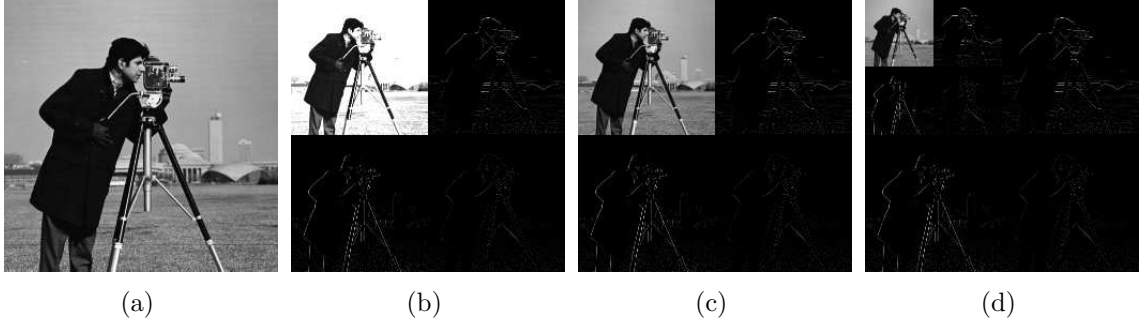


Figure 7: Schematic illustration of 2D wavelet transforms on an image. (a) is the original image, (b) is the original level-1 wavelet transformed image in (a), (c) is an amplitude adjustment of (b) to match the same brightness of (a), and (d) is the level-2 wavelet transform of the image in (a) with adjusted amplitude.

representation of \mathbf{V} , \mathbf{W} matrices,

$$cA = \mathbf{V}\mathbf{M}\mathbf{V}^T \quad (2.16)$$

$$cH = \mathbf{V}\mathbf{M}\mathbf{W}^T \quad (2.17)$$

$$cV = \mathbf{W}\mathbf{M}\mathbf{V}^T \quad (2.18)$$

$$cD = \mathbf{W}\mathbf{M}\mathbf{W}^T. \quad (2.19)$$

The inverse wavelet transform of the matrix $\begin{pmatrix} cA & cH \\ cV & cD \end{pmatrix}$ is

$$\begin{pmatrix} cA & cH \\ cV & cD \end{pmatrix} \xrightarrow{\text{IDWT}} (\mathbf{V}^T, \mathbf{W}^T) \begin{pmatrix} cA & cH \\ cV & cD \end{pmatrix} \begin{pmatrix} \mathbf{V} \\ \mathbf{W} \end{pmatrix} = \mathbf{M} \quad (2.20)$$

Combining the forward and inverse wavelet transforms, we can get an identity:

$$\mathbf{M} = (\mathbf{V}^T\mathbf{V} + \mathbf{W}^T\mathbf{W}) \mathbf{M} (\mathbf{V}^T\mathbf{V} + \mathbf{W}^T\mathbf{W}), \quad (2.21)$$

2.1 Haar Wavelet Transform

Recall that from Theorem 2.2, $\mathbf{V}^T\mathbf{V} + \mathbf{W}^T\mathbf{W} = \mathbf{I}$.

3 Wavelet Based Image Super-Resolution

3.1 Previous Work

Super-resolution is basically a process by which one gains spatial resolution in return for temporal bandwidth which refers to the availability of multiple non-redundant images of the same scene. Lukosz [Lukosz 1966, Lukosz 1967] was first to realize this process. The problem has been an active research area since the seminal work by Tsai and Huang [Tsai and Huang 1984] which considers the problem of reconstructing a resolution enhanced image from a sequence of low resolution (LR) images of a translated scene. Various algorithms have been developed in super-resolution and here we just categorize them into two main divisions - frequency domain and spatial domain.

3.1.1 Frequency Domain Methods

Frequency domain methods are utilized in a large class of SR algorithms. These methods are based on three fundamental principles:

1. the shifting property of the Fourier transform (FT)
2. the aliasing relationship between the continuous Fourier Transform (CFT) and the discrete Fourier transform (DFT)
3. the original scene is band-limited.

These properties allow the formulation of a system of equations relating the aliased DFT coefficients of the observed images to samples of the CFT of the unknown scene. These equations are solved yielding the frequency domain coefficients of the original

scene, which may then be recovered by inverse DFT. In this process, a linear system is formed and various least squares methods [Kim 1990, Tekalp 1992, Bose 1993, Kim and Su 1993] have been proposed to solve the linear system.

Techniques based on the multichannel sampling theorem [Brown 1981] have also been proposed [Ur and Gross 1992]. This technique is fundamentally a frequency domain method relying on the shift property of the Fourier transform to model the translation of the source image, even though they are implemented in the spatial domain.

Frequency domain SR methods provide the advantages of theoretical simplicity and low computational complexity. In addition, they are highly amenable to parallel implementation and exhibit an intuitive de-aliasing SR mechanism. Disadvantages include the limitation to global translational motion and space invariant degradation models (necessitated by the requirement for a Fourier domain analog of the spatial domain motion and degradation model) and limited ability for inclusion of spatial domain *a-priori* knowledge for regularization. Another limitation of the frequency domain method is the need to solve a linear system, no matter what kind of variational least squares method is used.

3.1.2 Spatial Domain Methods

Conjugate to frequency domain methods, the spatial domain methods are another large class of algorithms for reconstructing super-resolution images.

The iterated backprojection method is a widely used iterative method in the spatial domain. Given an SR estimate $\hat{\mathbf{z}}$ and the imaging model \mathbf{H} , it is possible to simulate the LR images $\hat{\mathbf{Y}}$ as $\hat{\mathbf{Y}} = \mathbf{H}\hat{\mathbf{z}}$. Iterated backprojection (IBP) procedures

update the estimate of the SR reconstruction by back-projecting the error between the j^{th} simulated LR image $\hat{\mathbf{Y}}^{(j)}$ and the observed LR images \mathbf{Y} via the backprojection operator \mathbf{H}^{BP} which apportions “blame” to pixels in the SR estimate $\hat{\mathbf{z}}^{(j)}$. Typically \mathbf{H}^{BP} approximates \mathbf{H}^{-1} . Algebraically,

$$\hat{\mathbf{z}}^{(j+1)} = \hat{\mathbf{z}}^{(j)} + \mathbf{H}^{BP} (\mathbf{Y} - \hat{\mathbf{Y}}^{(j)}) = \hat{\mathbf{z}}^{(j)} + \mathbf{H}^{BP} (\mathbf{Y} - \mathbf{H}\hat{\mathbf{z}}^{(j)}). \quad (3.1)$$

Eq. 3.1 is iterated until some error criterion dependent on \mathbf{Y} , $\hat{\mathbf{Y}}^{(j)}$ is minimized. Application of the IBP method may be found in [Irani and Peleg 1993]. IBP requires that the SR reconstruction match the observed data. Unfortunately, since SR is an ill-posed inverse problem, the SR reconstruction is not unique. In addition, inclusion of *a-priori* constraints is not easily achieved in the IBP method.

Stochastic methods (Bayesian in particular) are also applied to the image super-resolution problem. Stochastic methods, which treat the SR reconstruction as a statistical estimation problem, have rapidly gained prominence since they provide a powerful theoretical framework for the inclusion of *a-priori* constraints necessary for satisfactory solution of the ill-posed SR inverse problem. The observed data \mathbf{Y} , noise \mathbf{N} and SR image \mathbf{z} are assumed stochastic. The *Maximum A-Posteriori* (MAP) approach to estimate \mathbf{z} seeks the estimate $\hat{\mathbf{z}}_{\text{MAP}}$ for which the *a-posteriori* probability, $\Pr\{\mathbf{z}|\mathbf{Y}\}$ is a maximum. Since $\mathbf{Y} = \mathbf{H}\mathbf{z} + \mathbf{N}$, the likelihood function is determined by the PDF of the noise as $\Pr\{\mathbf{Y}|\mathbf{z}\} = f_N(\mathbf{Y} - \mathbf{H}\mathbf{z})$. It is common to use Markov random field (MRF) image models as the prior term $\Pr\{\mathbf{z}\}$. Under typical assumptions of Gaussian noise the prior may be chosen to ensure a convex optimization enabling the use of descent optimization procedures. Examples of the application of Bayesian methods to SR reconstruction may be found in [Schultz 1996] using a Huber MRF

and [Cheeseman 1996, Hardie 1997] with a Gaussian MRF.

Maximum likelihood (ML) estimation has also been applied to SR reconstruction [Tom 1994]. ML estimation is a special case of MAP estimation with no prior term. Since the inclusion of *a-priori* information is essential for the solution of ill-posed inverse problems, MAP estimation should be used in preference to ML.

A major advantage of the Bayesian framework is the direct inclusion of *a-priori* constraints on the solution, often as MRF priors which provide a powerful method for image modeling using (possibly non-linear) local neighbor interaction. MAP estimation with convex priors provides a globally convex optimization model, ensuring solution existence and uniqueness, and allowing application of efficient descent optimization methods. Simultaneous motion estimation and restoration is also possible [Hardie 1997]. The rich area of statistical estimation theory is directly applicable to stochastic SR reconstruction methods.

There are some other well established methods such as set theoretical reconstruction methods [Tom 1996, Patti 1997, Eren 1997], hybrid ML/MAP/POCS methods [Schultz 1996, Elad 1997], optimal and adaptive filtering and Tikhonov-Arsenin regulation [Patti 1998] etc. applied in image super-resolution.

Spatial domain SR reconstruction methods offer important advantages in terms of flexibility, but they are computationally more expensive and more complex than frequency domain methods.

In recent years, another class of algorithms based on wavelet transforms have been developed [Nguyen 2000, Bose 2004, Shen 2004, El-Khamy 2005, Chang 2006, Watanabe 2006].

Super-resolution involves two key steps - registration and reconstruction. Most of

3.2 Intuitive Algorithm

the proposed algorithms separate these two steps, and solve each step independently. On the other hand, some algorithms have been developed to couple these two steps and achieve some very good results [Chung 2006]. In this thesis, we are just focusing on the reconstruction step.

3.2 Intuitive Algorithm

In this section, an intuitive idea for wavelet based super-resolution is applied to 1D curves and 2D images and a schematic algorithm is also presented.

3.2.1 1D Curve

Suppose we have two low resolution 1D curves L_0 and L_1 at resolution n . For example, suppose the original image sample is known as S , of resolution $2n$, L_0 can be just the down sampling image of S . To get another non-trivial low resolution image L_1 , we can shift S one pixel forward and down sample. One of the most intuitive ideas to perform super-resolution is to superimpose L_0 and L_1 according to their known relative positions (R_0 and R_1), then average the shifted L_0 and L_1 to get an initial guess S_0 of the original curve S . It is obvious that S_0 is unlikely be the same as S . As a result, some updating schemes must be applied to refine the initial guess S_0 . Since S_0 is a guess or approximation of S , it contains more information than both L_0 and L_1 . It should be possible to extract some information about L_0 from S_0 . According to the relation between wavelet transforms and super-resolution, we can perform a Haar wavelet transform on S_0 to get the approximation A_0 and detail D_0 of S_0 . Recall the generation of L_0 is just the level-1 approximation of S . In other words, if we apply the wavelet transform to S , L_0 is the level-1 approximation, but the detail is unknown.

3.2 Intuitive Algorithm

Similarly, if we perform a wavelet transform on the one-pixel shifted S , we will get L_1 , again without the detail. Note that because L_0 is the exact approximation of S , and D_0 may contain some information about the lost detail, we can combine them together and perform the inverse wavelet transform to try to update S_0 . Combining the results from L_0 and L_1 , we can update S_0 to S_1 and continue the update until S_{i+1} is close enough to S_i .

The detailed algorithm of the above intuitive idea is sketched in Fig. 8 (a). The algorithm is applied to a simple example. As seen in Table 1, after about 50 steps, the result is quite close to the true image.

Table 1: Test of the intuitive algorithm on a simple 1D curve.

ITER	a_1	a_2	a_3	a_4	a_5	a_6	ITER	a_1	a_2	a_3	a_4	a_5	a_6
0	0.00	2.00	4.00	6.00	8.00	0.00	25	0.00	2.03	3.95	6.05	7.97	0.00
1	0.00	2.00	4.00	6.00	5.50	0.00	26	0.00	2.03	3.95	6.05	7.97	0.00
2	0.00	2.00	4.00	6.63	6.75	0.00	27	0.00	2.03	3.96	6.04	7.97	0.00
3	0.00	2.00	3.84	6.63	7.22	0.00	28	0.00	2.02	3.96	6.04	7.98	0.00
4	0.00	2.04	3.77	6.55	7.45	0.00	29	0.00	2.02	3.97	6.03	7.98	0.00
5	0.00	2.08	3.74	6.47	7.59	0.00	30	0.00	2.02	3.97	6.03	7.98	0.00
6	0.00	2.10	3.73	6.40	7.68	0.00	31	0.00	2.02	3.97	6.03	7.98	0.00
7	0.00	2.12	3.74	6.35	7.74	0.00	32	0.00	2.02	3.98	6.02	7.98	0.00
8	0.00	2.13	3.75	6.31	7.78	0.00	33	0.00	2.01	3.98	6.02	7.99	0.00
9	0.00	2.12	3.77	6.27	7.81	0.00	34	0.00	2.01	3.98	6.02	7.99	0.00
10	0.00	2.12	3.79	6.24	7.84	0.00	35	0.00	2.01	3.98	6.02	7.99	0.00
11	0.00	2.11	3.80	6.21	7.86	0.00	36	0.00	2.01	3.98	6.02	7.99	0.00
12	0.00	2.11	3.82	6.19	7.88	0.00	37	0.00	2.01	3.98	6.02	7.99	0.00
13	0.00	2.10	3.84	6.17	7.89	0.00	38	0.00	2.01	3.99	6.01	7.99	0.00
14	0.00	2.09	3.85	6.15	7.90	0.00	39	0.00	2.01	3.99	6.01	7.99	0.00
15	0.00	2.08	3.86	6.14	7.91	0.00	40	0.00	2.01	3.99	6.01	7.99	0.00
16	0.00	2.08	3.88	6.13	7.92	0.00	41	0.00	2.01	3.99	6.01	7.99	0.00
17	0.00	2.07	3.89	6.11	7.93	0.00	42	0.00	2.01	3.99	6.01	7.99	0.00
18	0.00	2.06	3.90	6.10	7.94	0.00	43	0.00	2.01	3.99	6.01	7.99	0.00
19	0.00	2.06	3.91	6.09	7.94	0.00	44	0.00	2.00	3.99	6.01	8.00	0.00
20	0.00	2.05	3.92	6.08	7.95	0.00	45	0.00	2.00	3.99	6.01	8.00	0.00
21	0.00	2.05	3.93	6.08	7.95	0.00	46	0.00	2.00	3.99	6.01	8.00	0.00
22	0.00	2.04	3.93	6.07	7.96	0.00	47	0.00	2.00	3.99	6.01	8.00	0.00
23	0.00	2.04	3.94	6.06	7.96	0.00	48	0.00	2.00	4.00	6.00	8.00	0.00
24	0.00	2.03	3.94	6.06	7.97	0.00							

3.2 Intuitive Algorithm

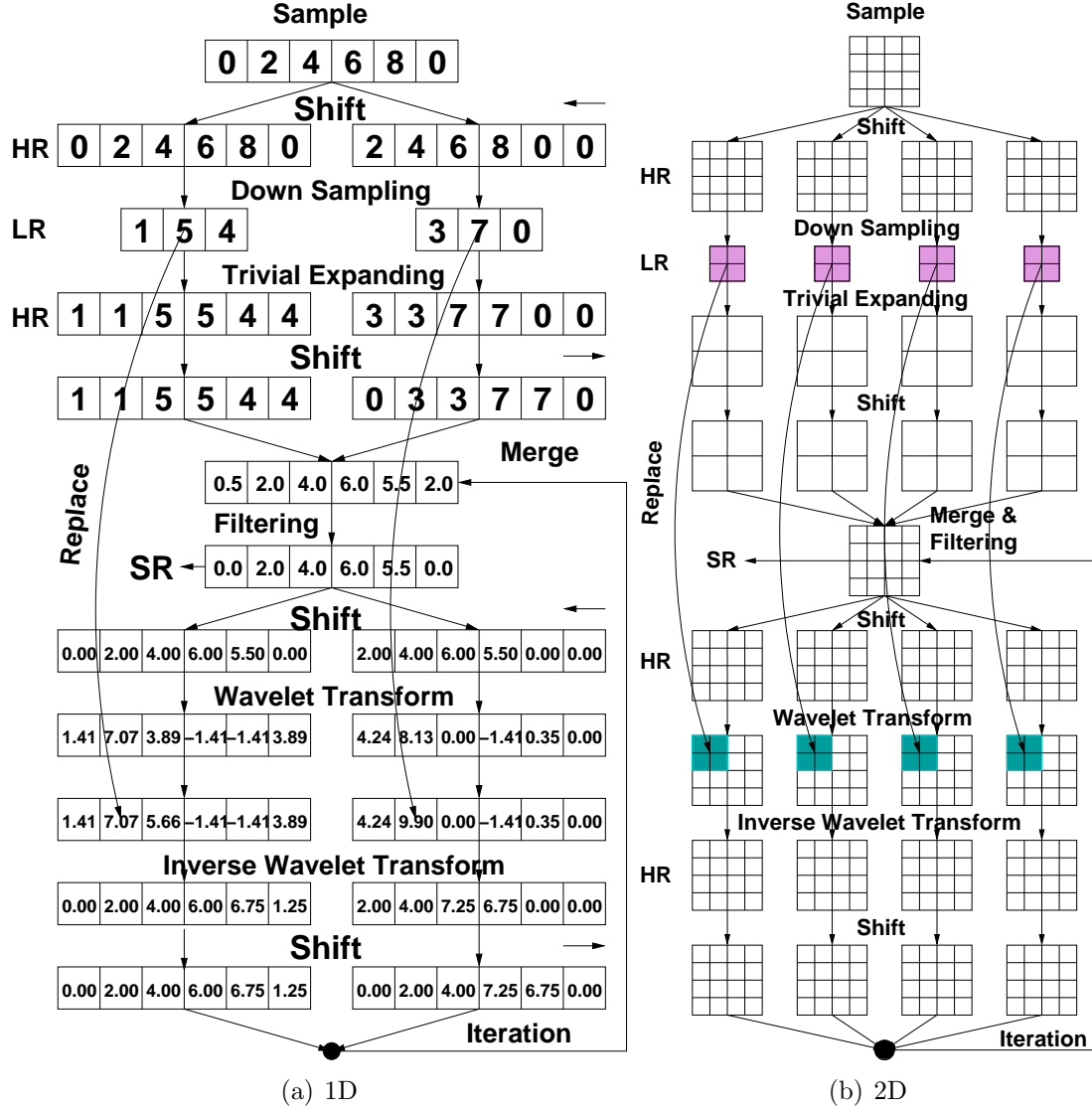


Figure 8: Illustration of intuitive wavelet based super-resolution algorithm applied to a 1D example (left) and a 2D example (right).

3.2 Intuitive Algorithm

If we define the error between S_i and S as $\|S_i - S\|$, the convergence of the algorithm for the above simple example can be shown in Fig. 9. The error is steadily decreasing as seen from Fig. 9, but about 60 steps are needed for such a simple example and it is difficult to provide the convergence guarantee for a more general case.

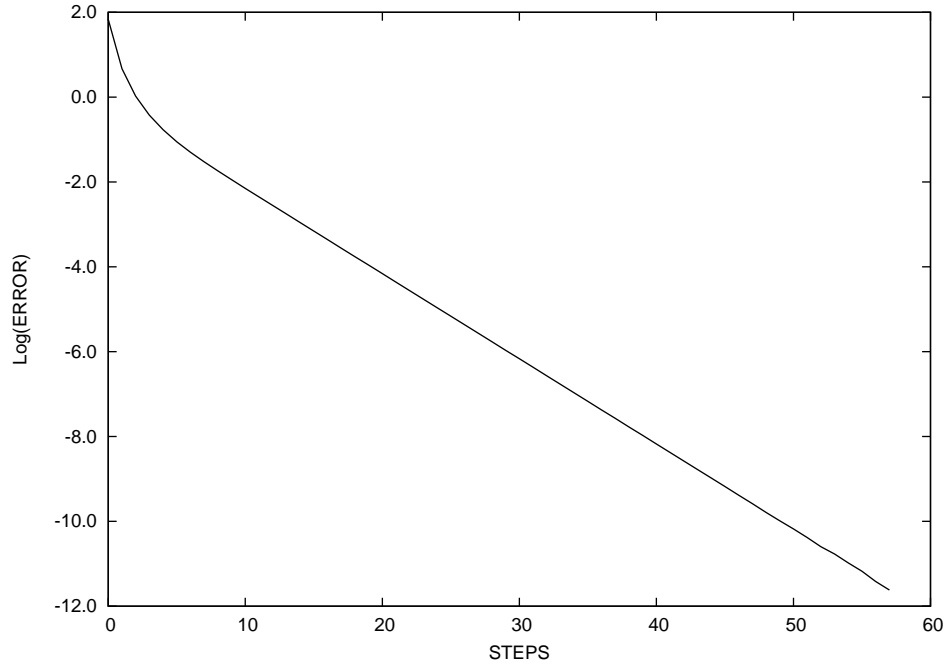


Figure 9: 1D level-1 up super-resolution error convergence.

Without worrying about the convergence, the algorithm for a 1D curve can be easily extended to multilevel super-resolution and it will be described in later sections.

3.2.2 2D Images

The intuitive idea based algorithm works in the 1D case, and the simple idea can be easily extended to higher dimensions with little modification. As a result, a similar algorithm for the 2D case is sketched in Fig. 8 (b). In addition, the algorithm for

3.2 Intuitive Algorithm

super-resolution is not limited to level-1 up, it is suitable for multilevel up super-resolution as shown in the following examples.

The first example we applied the generalized algorithm to is the “cameraman”. As shown in Fig. 10 (a), the original image has a resolution of 256×256 . Four low resolution images were generated based on the original image. Fig. 10 (b) is one of the four low resolution images.

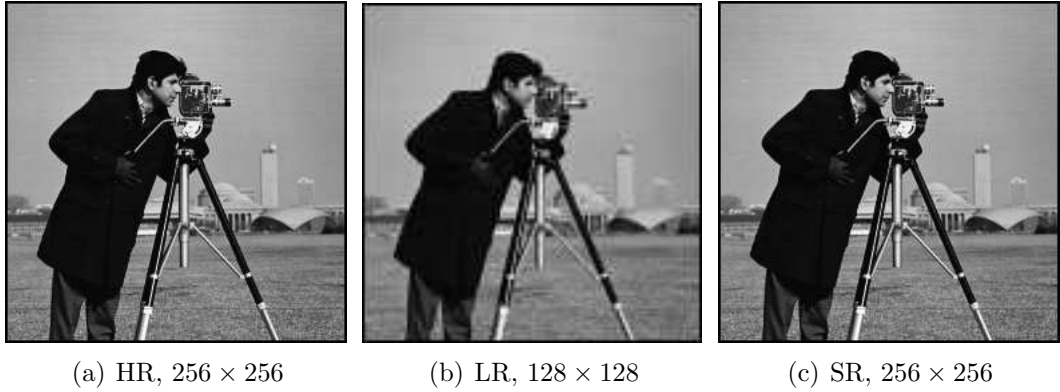


Figure 10: 2D level-1 up super-resolution. (a) is the original high resolution image, (b) is one of the four low resolution images generated from (a), and (c) is the super-resolution image after about 110 iterations.

After about 110 iterations, a super-resolution image at resolution 256×256 is obtained and shown in Fig. 10 (c) and the error convergence is shown in Fig. 11. As can be seen, the convergence rate is worse than that in the 1D case, but the error keeps on decreasing with more and more iteration steps.

Further, the same algorithm is performed to the level-2 up super-resolution as shown in Fig. 12. In the 2D level-2 case, 16 low resolution images are needed for recovering the original high resolution image.

The error convergence rate for 2D level-2 up case is shown in Fig. 13. This time the algorithm converges more slowly than the level-1 up case.

3.2 Intuitive Algorithm

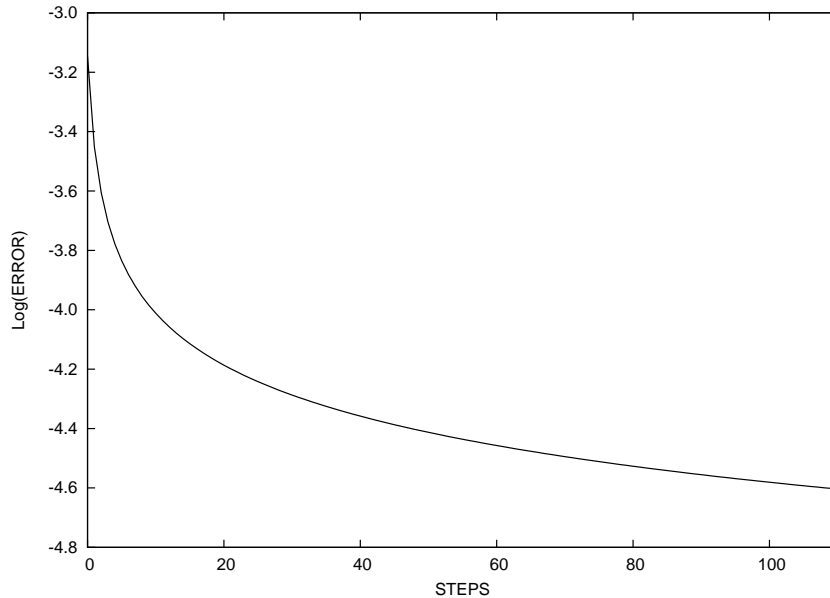


Figure 11: 2D level-1 up super-resolution error convergence.



Figure 12: 2D level-2 up super-resolution. (a) is the original high resolution image, (b) is one of the 16 low resolution images generated from (a), and (c) is the super-resolution image after about 600 iterations.

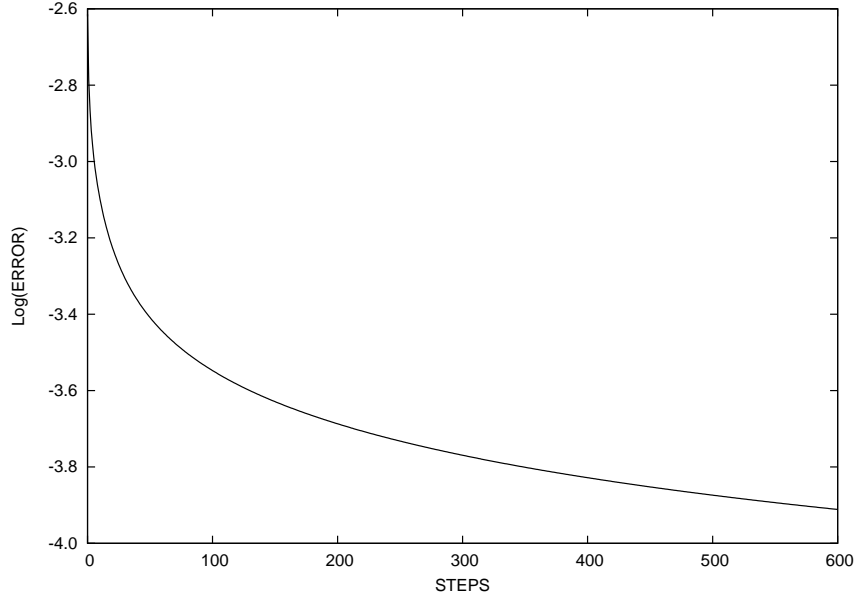


Figure 13: 2D level-2 up super-resolution error convergence.

3.3 Matrix Form Algorithm

Although the intuitive algorithm works well for both 1D and 2D images, the problem remains to understand its convergence behavior. In addition, the algorithm is loosely organized and not in a simple and elegant mathematical form, though the idea is very simple. In this section, we will prove some results related to the wavelet and scaling matrices, show that the intuitive approach in the previous section is equivalent to a matrix form algorithm, and prove convergence of the algorithm. Further, it can be shown that the intuitive algorithm or the matrix form algorithm is just a special form of the “Landweber Iteration” method to solve the linear systems. However, we will not stop there - a one step solution for the 1D super-resolution problem is proposed which totally avoids the iteration. Based on the equivalence of the 1D intuitive algorithm and the 1D matrix form algorithm, the 2D matrix form algorithm is also developed for both level-1 and multilevel up super-resolution.

3.3.1 1D and Level-1 Up Super-Resolution

Before presenting some theorems, some definitions are needed.

Definition Let $\mathbf{F} \in \Re^{n \times n}$, $k \geq 0$ and

$$F_{i,j} = \begin{cases} 1, & i = j \text{ and } (1 + k \leq i \leq n - k) \\ 0, & \text{otherwise.} \end{cases}$$

Then, \mathbf{F} is said to be a filter matrix of order k . For simplicity, a filter matrix of order 1 is simply referenced as a filter matrix.

Proposition 3.1 *An order 0 filter matrix is an identity matrix.*

Definition A square matrix \mathbf{R} is said to be an order d **up-shift matrix** if

$$R_{i,j} = \begin{cases} 1, & j = i + d \\ 0, & \text{else.} \end{cases}$$

For example,

$$\mathbf{R} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

3.3 Matrix Form Algorithm

is an order 1 up-shift matrix. It acts on a vector $v = (1 \ 2 \ 3 \ 4)^T$ as

$$\mathbf{R}v = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix} = \begin{pmatrix} 2 \\ 3 \\ 4 \\ 0 \end{pmatrix}.$$

It is obvious that all the elements in v are shifted one position up and this is the reason that matrix \mathbf{R} is called an up-shift matrix. Note that if d is negative, \mathbf{R} actually shifts all the elements in a vector down, and it is more properly called a *down-shift matrix*. For the purpose of generality, these kind of matrices are still called *up-shift matrices* but with negative order.

Proposition 3.2 *An order 0 up-shift matrix is an identity matrix.*

Proposition 3.3 *If $\mathbf{R} \in \mathfrak{R}^{n \times n}$ is an up-shift matrix of order j ($j \geq 0$) and $\mathbf{F} \in \mathfrak{R}^{n \times n}$ is a filter matrix of order k ($k \geq j$), then $\mathbf{F}\mathbf{R}^T\mathbf{R} = \mathbf{F}$.*

Theorem 3.4 (1DL1) *Let $f \in \mathfrak{R}^n$ (n is even), and $f_1 = f_n = 0$. $\mathbf{F} \in \mathfrak{R}^{n \times n}$ is an order 1 filter matrix. $\mathbf{R}_0, \mathbf{R}_1 \in \mathfrak{R}^{n \times n}$ are up-shift matrices of order 0 and 1. $\mathbf{V} \in \mathfrak{R}^{\frac{n}{2} \times n}$ is the scaling matrix and $\mathbf{W} \in \mathfrak{R}^{\frac{n}{2} \times n}$ is the wavelet matrix. Denote*

1. $f^{(0)} = \frac{1}{2}\mathbf{F}(\mathbf{R}_0^T\mathbf{V}^T\mathbf{V}\mathbf{R}_0 + \mathbf{R}_1^T\mathbf{V}^T\mathbf{V}\mathbf{R}_1)f$
2. $f^{(i)} = f^{(0)} + \frac{1}{2}\mathbf{F}(\mathbf{R}_0^T\mathbf{W}^T\mathbf{W}\mathbf{R}_0 + \mathbf{R}_1^T\mathbf{W}^T\mathbf{W}\mathbf{R}_1)f^{(i-1)}, \forall i \geq 1.$

Then,

1. $f = f^{(0)} + \frac{1}{2}\mathbf{F}(\mathbf{R}_0^T\mathbf{W}^T\mathbf{W}\mathbf{R}_0 + \mathbf{R}_1^T\mathbf{W}^T\mathbf{W}\mathbf{R}_1)f,$

$$2. \lim_{n \rightarrow \infty} f^{(i)} = f.$$

Proof 1. Because $f \in \mathfrak{R}^n$ (n is even), and $f_1 = f_n = 0$, it follows that

$$\mathbf{F}f = f.$$

Further, \mathbf{V} is a scaling matrix and \mathbf{W} is a wavelet matrix, according to Theorem 2.2 which says that $\mathbf{V}^T \mathbf{V} + \mathbf{W}^T \mathbf{W} = \mathbf{I}$, therefore,

$$\begin{aligned} \mathbf{F}[\mathbf{R}_0^T (\mathbf{V}^T \mathbf{V} + \mathbf{W}^T \mathbf{W}) \mathbf{R}_0 + \mathbf{R}_1^T (\mathbf{V}^T \mathbf{V} + \mathbf{W}^T \mathbf{W}) \mathbf{R}_1] \\ = \mathbf{F}(\mathbf{R}_0^T \mathbf{R}_0) + \mathbf{F}(\mathbf{R}_1^T \mathbf{R}_1) = 2\mathbf{F}, \end{aligned}$$

therefore,

$$\frac{1}{2} \mathbf{F}[\mathbf{R}_0^T (\mathbf{V}^T \mathbf{V} + \mathbf{W}^T \mathbf{W}) \mathbf{R}_0] + \frac{1}{2} \mathbf{F}[\mathbf{R}_1^T (\mathbf{V}^T \mathbf{V} + \mathbf{W}^T \mathbf{W}) \mathbf{R}_1] = \mathbf{F}.$$

So,

$$\frac{1}{2} \mathbf{F}[\mathbf{R}_0^T (\mathbf{V}^T \mathbf{V} + \mathbf{W}^T \mathbf{W}) \mathbf{R}_0] f + \frac{1}{2} \mathbf{F}[\mathbf{R}_1^T (\mathbf{V}^T \mathbf{V} + \mathbf{W}^T \mathbf{W}) \mathbf{R}_1] f = \mathbf{F}f = f.$$

Since

$$f^{(0)} = \frac{1}{2} \mathbf{F} (\mathbf{R}_0^T \mathbf{V}^T \mathbf{V} \mathbf{R}_0 + \mathbf{R}_1^T \mathbf{V}^T \mathbf{V} \mathbf{R}_1) f,$$

we see that,

$$f = f^{(0)} + \frac{1}{2} \mathbf{F} (\mathbf{R}_0^T \mathbf{W}^T \mathbf{W} \mathbf{R}_0 + \mathbf{R}_1^T \mathbf{W}^T \mathbf{W} \mathbf{R}_1) f. \quad \blacksquare$$

3.3 Matrix Form Algorithm

2. Denote $\epsilon^{(i)} = f - f^{(i)}$, $\forall i \geq 0$.

Since

$$f = f^{(0)} + \frac{1}{2} \mathbf{F} (\mathbf{R}_0^T \mathbf{W}^T \mathbf{W} \mathbf{R}_0 + \mathbf{R}_1^T \mathbf{W}^T \mathbf{W} \mathbf{R}_1) f$$

and

$$f^{(i)} = f^{(0)} + \frac{1}{2} \mathbf{F} (\mathbf{R}_0^T \mathbf{W}^T \mathbf{W} \mathbf{R}_0 + \mathbf{R}_1^T \mathbf{W}^T \mathbf{W} \mathbf{R}_1) f^{(i-1)}, \forall i \geq 1.$$

We obtain,

$$\epsilon^{(i)} = \frac{1}{2} \mathbf{F} (\mathbf{R}_0^T \mathbf{W}^T \mathbf{W} \mathbf{R}_0 + \mathbf{R}_1^T \mathbf{W}^T \mathbf{W} \mathbf{R}_1) \epsilon^{(i-1)}.$$

Because

$$\begin{aligned} \frac{1}{2} \|\mathbf{F} (\mathbf{R}_0^T \mathbf{W}^T \mathbf{W} \mathbf{R}_0 + \mathbf{R}_1^T \mathbf{W}^T \mathbf{W} \mathbf{R}_1)\| & \\ & \leq \frac{1}{2} (\|\mathbf{F} \mathbf{R}_0^T \mathbf{W}^T \mathbf{W} \mathbf{R}_0\| + \|\mathbf{F} \mathbf{R}_1^T \mathbf{W}^T \mathbf{W} \mathbf{R}_1\|) \end{aligned}$$

and

$$\|\mathbf{F} \mathbf{R}_0^T \mathbf{W}^T \mathbf{W} \mathbf{R}_0\| \leq 1 \text{ and } \|\mathbf{F} \mathbf{R}_1^T \mathbf{W}^T \mathbf{W} \mathbf{R}_1\| \leq 1,$$

it follows that,

$$\frac{1}{2} \|\mathbf{F} (\mathbf{R}_0^T \mathbf{W}^T \mathbf{W} \mathbf{R}_0 + \mathbf{R}_1^T \mathbf{W}^T \mathbf{W} \mathbf{R}_1)\| \leq 1.$$

Since

$$\mathbf{F} \mathbf{R}_0^T \mathbf{W}^T \mathbf{W} \mathbf{R}_0 \neq \mathbf{F} \mathbf{R}_1^T \mathbf{W}^T \mathbf{W} \mathbf{R}_1,$$

it follows that,

$$\frac{1}{2} \|\mathbf{F} (\mathbf{R}_0^T \mathbf{W}^T \mathbf{W} \mathbf{R}_0 + \mathbf{R}_1^T \mathbf{W}^T \mathbf{W} \mathbf{R}_1)\| < 1.$$

3.3 Matrix Form Algorithm

Therefore,

$$\|\epsilon^{(i)}\| < \|\epsilon^{(i-1)}\|.$$

So,

$$\lim_{i \rightarrow \infty} \epsilon^{(i)} = 0.$$

Finally,

$$\lim_{i \rightarrow \infty} f^{(i)} = f. \quad \blacksquare$$

Theorem 3.4 provides an updating scheme to recover the original image and also guarantees the convergence of the algorithm. This theorem also simplifies the steps in performing super-resolution, but it is just an abstraction of the intuitive algorithm, and in the following subsection we will show that it is equivalent to the intuitive algorithm.

3.3.2 Relation Between the Intuitive and Matrix Form Algorithm

To demonstrate the equivalence of the matrix form algorithm (Theorem 3.4) and the intuitive algorithm (Fig. 8), we need more definitions.

Definition An $\frac{n}{2} \times n$ matrix \mathbf{D} is called a *down sampling matrix* if $\mathbf{D} = \frac{1}{\sqrt{2}}\mathbf{V}$ where \mathbf{V} is the scaling matrix defined in Eq. 2.7.

For example,

$$\mathbf{D} = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

3.3 Matrix Form Algorithm

is a down sampling matrix. When \mathbf{D} acts on a vector

$$v = \begin{pmatrix} 0 \\ 2 \\ 4 \\ 6 \\ 8 \\ 0 \end{pmatrix}$$

as

$$\begin{pmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 2 \\ 4 \\ 6 \\ 8 \\ 0 \end{pmatrix},$$

it gives a shorter vector

$$l = \begin{pmatrix} 1 \\ 5 \\ 4 \end{pmatrix},$$

whose elements are just the pairwise average of the elements of v .

Definition An $\frac{n}{2} \times n$ matrix \mathbf{T} is called a *trivial expanding matrix* if $\mathbf{T} = \sqrt{2}\mathbf{V}^T$ where \mathbf{V} is the scaling matrix as defined in Eq. 2.7.

For example,

$$\mathbf{T} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

is a trivial expanding matrix. When \mathbf{T} acts on a vector

$$l = \begin{pmatrix} 1 \\ 5 \\ 4 \end{pmatrix}$$

as

$$\begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 5 \\ 4 \end{pmatrix},$$

it gives a longer vector

$$l = \begin{pmatrix} 1 \\ 1 \\ 5 \\ 5 \\ 4 \\ 4 \end{pmatrix}.$$

Proposition 3.5 *The trivial expanding matrix is twice of the transpose of the down-sampling matrix, or $\mathbf{T} = 2\mathbf{D}^T$.*

Proof Since

$$\mathbf{D} = \frac{1}{\sqrt{2}}\mathbf{V} \text{ and } \mathbf{T} = \sqrt{2}\mathbf{V}^T,$$

it clearly follows that

$$\mathbf{T} = 2\mathbf{D}^T. \quad \blacksquare$$

Theorem 3.6 *The intuitive algorithm shown in Fig. 8 is equivalent to the updating scheme as:*

1. $f^{(0)} = \frac{1}{2}\mathbf{F} (\mathbf{R}_0^T \mathbf{V}^T \mathbf{V} \mathbf{R}_0 + \mathbf{R}_1^T \mathbf{V}^T \mathbf{V} \mathbf{R}_1) f$
2. $f^{(i)} = f^{(0)} + \frac{1}{2}\mathbf{F} (\mathbf{R}_0^T \mathbf{W}^T \mathbf{W} \mathbf{R}_0 + \mathbf{R}_1^T \mathbf{W}^T \mathbf{W} \mathbf{R}_1) f^{(i-1)}, \forall i \geq 1.$

as in Theorem 3.4.

Proof The proof will just work through all the steps in the intuitive algorithm and put each step in the matrix formation. In the end, we will show that the two algorithms are the same.

1. **Shift Forward**

$h^{(0)} = \mathbf{R}_0 f$ and $h^{(1)} = \mathbf{R}_1 f$ where \mathbf{R}_0 and \mathbf{R}_1 are *up-shift matrices*.

2. **Down Sampling**

$l^{(0)} = \mathbf{D}h^{(0)}$ and $l^{(1)} = \mathbf{D}h^{(1)}$ where \mathbf{D} is the *down sampling matrix*. Plug in $h^{(0)}$ and $h^{(1)}$ to obtain $l^{(0)} = \mathbf{D}\mathbf{R}_0 f$ and $l^{(1)} = \mathbf{D}\mathbf{R}_1 f$.

3. **Trivial Expanding**

$$h^{(0)} = \mathbf{T}l^{(0)} = 2\mathbf{D}^T l^{(0)} = 2\mathbf{D}^T \mathbf{D}\mathbf{R}_0 f$$

$$h^{(1)} = \mathbf{T}l^{(1)} = 2\mathbf{D}^T l^{(1)} = 2\mathbf{D}^T \mathbf{D}\mathbf{R}_1 f$$

4. **Shift Reverse**

$$h^{(0)} = \mathbf{R}_0^T \mathbf{T}l^{(0)} = 2\mathbf{R}_0^T \mathbf{D}^T l^{(0)} = 2\mathbf{R}_0^T \mathbf{D}^T \mathbf{D}\mathbf{R}_0 f$$

$$h^{(1)} = \mathbf{R}_1^T \mathbf{T}l^{(1)} = 2\mathbf{R}_1^T \mathbf{D}^T l^{(1)} = 2\mathbf{R}_1^T \mathbf{D}^T \mathbf{D}\mathbf{R}_1 f$$

Because $\mathbf{D} = \frac{1}{\sqrt{2}}\mathbf{V}$, we get

$$2\mathbf{D}^T \mathbf{D} = \mathbf{V}^T \mathbf{V}. \text{ Thus}$$

$$h^{(0)} = \mathbf{R}_0^T \mathbf{V}^T \mathbf{V}\mathbf{R}_0 f, \text{ and } h^{(1)} = \mathbf{R}_1^T \mathbf{V}^T \mathbf{V}\mathbf{R}_1 f.$$

5. **Merge**

$$f^{(0)} = \frac{1}{2} (h^{(0)} + h^{(1)}) = \frac{1}{2} (\mathbf{R}_0^T \mathbf{V}^T \mathbf{V}\mathbf{R}_0 + \mathbf{R}_1^T \mathbf{V}^T \mathbf{V}\mathbf{R}_1) f.$$

6. **Filtering** $\boxed{f^{(0)} = \mathbf{F}f^{(0)} = \frac{1}{2}\mathbf{F} (\mathbf{R}_0^T \mathbf{V}^T \mathbf{V}\mathbf{R}_0 + \mathbf{R}_1^T \mathbf{V}^T \mathbf{V}\mathbf{R}_1) f}.$

7. **Shift Foward**

$$h^{(0)} = \mathbf{R}_0 f^{(0)}, h^{(1)} = \mathbf{R}_1 f^{(0)}.$$

8. **Wavelet Transform**

$$a^{(0)} = \mathbf{V}h^{(0)}, \text{ and } a^{(1)} = \mathbf{V}h^{(1)}.$$

$$d^{(0)} = \mathbf{W}h^{(0)} = \mathbf{W}\mathbf{R}_0f^{(0)}, \text{ and}$$

$$d^{(1)} = \mathbf{W}h^{(1)} = \mathbf{W}\mathbf{R}_1f^{(0)}.$$

9. **Replace**

$$a^{(0)} = \sqrt{2}l^{(0)} = \sqrt{2}\mathbf{D}\mathbf{R}_0f \text{ and } a^{(1)} = \sqrt{2}l^{(1)} = \sqrt{2}\mathbf{D}\mathbf{R}_1f.$$

10. **Inverse Wavelet Transform**

$$h^{(0)} = \mathbf{V}^T a^{(0)} + \mathbf{W}^T d^{(0)} = \mathbf{V}^T \sqrt{2}\mathbf{D}\mathbf{R}_0f + \mathbf{W}^T \mathbf{W}\mathbf{R}_0f^{(0)} = \mathbf{V}^T \mathbf{V}\mathbf{R}_0f + \mathbf{W}^T \mathbf{W}\mathbf{R}_0f^{(0)}$$

$$h^{(1)} = \mathbf{V}^T a^{(1)} + \mathbf{W}^T d^{(1)} = \mathbf{V}^T \sqrt{2}\mathbf{D}\mathbf{R}_1f + \mathbf{W}^T \mathbf{W}\mathbf{R}_1f^{(0)} = \mathbf{V}^T \mathbf{V}\mathbf{R}_1f + \mathbf{W}^T \mathbf{W}\mathbf{R}_1f^{(0)}$$

11. **Shift Reverse**

$$h^{(0)} = \mathbf{R}_0^T h^{(0)} = \mathbf{R}_0^T \mathbf{V}^T \mathbf{V}\mathbf{R}_0f + \mathbf{R}_0^T \mathbf{W}^T \mathbf{W}\mathbf{R}_0f^{(0)}$$

$$h^{(1)} = \mathbf{R}_1^T h^{(1)} = \mathbf{R}_1^T \mathbf{V}^T \mathbf{V}\mathbf{R}_1f + \mathbf{R}_1^T \mathbf{W}^T \mathbf{W}\mathbf{R}_1f^{(0)}$$

$$\text{Denote } \mathbf{A}_1 = \mathbf{R}_0^T \mathbf{V}^T \mathbf{V}\mathbf{R}_0 \text{ and } \mathbf{A}_2 = \mathbf{R}_1^T \mathbf{V}^T \mathbf{V}\mathbf{R}_1$$

$$\mathbf{B}_1 = \mathbf{R}_0^T \mathbf{W}^T \mathbf{W}\mathbf{R}_0 \text{ and } \mathbf{B}_2 = \mathbf{R}_1^T \mathbf{W}^T \mathbf{W}\mathbf{R}_1,$$

then

$$h^{(0)} = \mathbf{A}_1f + \mathbf{B}_1f^{(0)} \text{ and } h^{(1)} = \mathbf{A}_2f + \mathbf{B}_2f^{(0)}$$

12. **Merge**

$$f^{(1)} = \frac{1}{2} (h^{(0)} + h^{(1)}) = \frac{1}{2} [(\mathbf{A}_1 + \mathbf{A}_2) f + (\mathbf{B}_1 + \mathbf{B}_2) f^{(0)}].$$

13. **Filtering**

$$f^{(1)} = \mathbf{F}f^{(1)} = \frac{1}{2}\mathbf{F} [(\mathbf{A}_1 + \mathbf{A}_2) f + (\mathbf{B}_1 + \mathbf{B}_2) f^{(0)}]$$

...

$$f^{(i)} = \frac{1}{2}\mathbf{F} [(\mathbf{A}_1 + \mathbf{A}_2) f + (\mathbf{B}_1 + \mathbf{B}_2) f^{(i-1)}]$$

$$\therefore \boxed{f^{(i)} = f^{(0)} + \frac{1}{2}\mathbf{F} (\mathbf{R}_0^T \mathbf{W}^T \mathbf{W} \mathbf{R}_0 + \mathbf{R}_1^T \mathbf{W}^T \mathbf{W} \mathbf{R}_1) f^{(i-1)}, \forall i \geq 1}. \quad \blacksquare$$

Since the intuitive idea is equivalent to the matrix form in the 1D case, and the intuitive algorithm is a general algorithm for both 1D and 2D image super-resolution, we derive the 2D matrix form algorithm from the intuitive idea in later sections without proof. Before presenting the general 2D matrix algorithm, we will show that the matrix form algorithm is indeed a special form of the *Landweber Iteration* algorithm for solving linear systems.

3.3.3 Landweber Iteration

The Landweber iteration method [Goldstein 1964, Baker 1977] is a well known iteration method to solve the linear systems $Ax = b$, and it is also well known for its slow convergence rate. Here we show that the 1D matrix form algorithm finally unites with the Landweber iteration for linear systems.

Proposition 3.7 *The 1D level-1 super-resolution algorithm is equivalent to the Landweber Iteration ($\omega = 1$) with projection \mathbf{F} , where \mathbf{F} is a order 1 filter matrix.*

Proof The Landweber iteration with projection for the linear system $Ax = b$ has the form

$$x^k = \mathbf{P}(x^{k-1} + \omega A^T r^{k-1}), \quad k = 1, 2, \dots,$$

where $r^k = b - Ax^k$. In the special case, let $\mathbf{P} = \mathbf{F}$ and $\omega = 1$, then

$$x^k = \mathbf{F}(x^{k-1} + A^T(b - Ax^{k-1})) = \mathbf{F}(x^{k-1} + A^T b - A^T Ax^{k-1}) = \mathbf{F}((I - A^T A)x^{k-1} + A^T Ax).$$

3.3 Matrix Form Algorithm

In the super-resolution case, A for the linear system is just $\begin{pmatrix} DR_0 \\ DR_1 \end{pmatrix}$ and $x = f$.
therefore,

$$A^T A = (R_0^T D^T, R_1^T D^T) \begin{pmatrix} DR_0 \\ DR_1 \end{pmatrix} = R_0^T D^T D R_0 + R_1^T D^T D R_1.$$

Because, $D^T D = \frac{1}{2} V^T V$, we see that, $A^T A = \frac{1}{2} (R_0^T V^T V R_0 + R_1^T V^T V R_1)$, and
thus,

$$f^{(0)} = \mathbf{F} A^T A f.$$

Furthermore,

$$\mathbf{F}(I - A^T A) = \frac{1}{2} \mathbf{F}(I - R_0^T V^T V R_0) + \frac{1}{2} \mathbf{F}(I - R_1^T V^T V R_1)$$

and

$$\mathbf{F} = \mathbf{F} R_0^T R_0 = \mathbf{F} R_1^T R_1,$$

so,

$$\begin{aligned} \mathbf{F}(I - A^T A) &= \frac{1}{2} \mathbf{F}(R_0^T (I - V^T V) R_0) + \frac{1}{2} \mathbf{F}(R_1^T (I - V^T V) R_1) \\ &= \frac{1}{2} \mathbf{F}(R_0^T W^T W R_0 + R_1^T W^T W R_1). \end{aligned}$$

Thus,

$$f^{(i)} = \mathbf{F} A^T A f + \mathbf{F}(I - A^T A) f^{(i-1)}. \quad \blacksquare$$

3.3.4 1D and Multilevel up Super-Resolution

In Section 3.3.1, the algorithm for 1D level-1 up super-resolution is developed and here we will generalize the algorithm from level-1 to multilevel up super-resolution. The generalization is based on the equivalence of the 1D level-1 up matrix form algorithm and the intuitive algorithm.

Definition In $\Re^{n \times n}$, define $\mathbf{A}_1 = \mathbf{V}_1$ and $\mathbf{A}_k = \mathbf{V}_k \mathbf{A}_{k-1}$ ($k \geq 2$), where \mathbf{V}_k is the level- k scaling matrix in $\Re^{n \times n}$. Similarly, define $\mathbf{B}_1 = \mathbf{W}_1$ and $\mathbf{B}_k = \mathbf{W}_k \mathbf{A}_{k-1}$ ($k \geq 2$), where \mathbf{W}_k is the level- k wavelet matrix in $\Re^{n \times n}$. Also, define $\mathbf{D}_k = \sum_{i=1}^k (\mathbf{B}_i^T \mathbf{B}_i)$

Proposition 3.8 In $\Re^{m \times m}$, ($m = 2^n$), $\mathbf{I}_m = \mathbf{A}_k^T \mathbf{A}_k + \mathbf{D}_k$, for all $1 \leq k \leq n$.

Proof According to Theorem 2.2,

$$\mathbf{I}_m = \mathbf{V}_1^T \mathbf{V}_1 + \mathbf{W}_1^T \mathbf{W}_1 = \mathbf{A}_1^T \mathbf{A}_1 + \mathbf{D}_1,$$

and

$$\mathbf{I}_{m/2} = \mathbf{V}_2^T \mathbf{V}_2 + \mathbf{W}_2^T \mathbf{W}_2.$$

Therefore,

$$\mathbf{V}_1^T \mathbf{V}_1 = \mathbf{V}_1^T \mathbf{I}_{m/2} \mathbf{V}_1 = \mathbf{V}_1^T (\mathbf{V}_2^T \mathbf{V}_2 + \mathbf{W}_2^T \mathbf{W}_2) \mathbf{V}_1,$$

and thus,

$$\mathbf{I}_m = \mathbf{A}_2^T \mathbf{A}_2 + \mathbf{D}_2.$$

The above steps can be continued for n times, and finally,

$$\mathbf{I}_m = \mathbf{A}_k^T \mathbf{A}_k + \mathbf{D}_k, \text{ for all } 1 \leq k \leq n. \quad \blacksquare$$

3.3 Matrix Form Algorithm

Theorem 3.9 *Let $\mathbf{R}_0, \mathbf{R}_1, \dots, \mathbf{R}_{k-1} \in \mathfrak{R}^{n \times n}$ be order $0, 1, \dots, k-1$ up-shift matrices, and let \mathbf{A}_l and \mathbf{D}_l be defined as above. Let $\mathbf{F} \in \mathfrak{R}^{n \times n}$ be a filtering matrix with k zeros in both ends of its diagonal. Then the algorithm for general level- l up super-resolution for $f \in \mathfrak{R}^n$ can be constructed as following:*

1. $f^{(0)} = \frac{1}{k} \mathbf{F} \left(\sum_{i=1}^k \mathbf{R}_i^T \mathbf{A}_l^T \mathbf{A}_l \mathbf{R}_i \right) f$
2. $f^{(s)} = f^{(0)} + \frac{1}{k} \mathbf{F} \left(\sum_{i=1}^k \mathbf{R}_i^T \mathbf{D}_l \mathbf{R}_i \right) f^{(s-1)}$

MATLAB code to test level-2 up super-resolution is present below:

```
n = 64;

f = rand(n,1);

m = 4;
f(1:m) = 0;
f(n-m+1:n) = 0;

F = ones(n,1);
F(1:m) = 0;
F(n-m+1:n) = 0;

% define the shift
R0 = diag(ones(1,n));
R1 = circshift(R0,[-1,0]); R1(n,1) = 0;
R2 = circshift(R1,[-1,0]); R2(n,2) = 0;
R3 = circshift(R2,[-1,0]); R3(n,3) = 0;

clear V1;
clear W1;
foo = [1 1 zeros(1,n-2)];
bar = [1 -1 zeros(1,n-2)];
foo = foo/sqrt(2);
bar = bar/sqrt(2);

for i = 1:n/2
    V1(i,:) = foo; foo = circshift(foo,[0,2]);
```

3.3 Matrix Form Algorithm

```

    W1(i,:) = bar; bar = circshift(bar,[0,2]);
end

V2 = V1(1:n/4,1:n/2);
W2 = W1(1:n/4,1:n/2);

A1 = V1;    D1 = W1;
A2 = V2*A1; D2 = W2*A1;

BA = A2'*A2; BD = D2'*D2 + D1'*D1;

f0 = (R0'*BA*R0 + R1'*BA*R1 + R2'*BA*R2 + R3'*BA*R3)*f.*F/4;
clear error;

pf = f0;
for i = 1:1000
    uf = (R0'*BD*R0 + R1'*BD*R1 + R2'*BD*R2 + R3'*BD*R3)*pf.*F/4;
    nf = f0 + uf;
    pf = nf;
    error(i) = norm(f-nf)/sqrt(n);
end

```

3.3.5 One Step Solution in the 1D Case

Mathematically, it is beautiful to finally rewrite the matrix form algorithm as a Landweber iteration, but a great simple algorithm will be buried under the well known “Landweber Iteration” form. After reexamining the original matrix form of the algorithm as shown in Theorem 3.4, we find the following one step solution for 1D level-1 up super-resolution.

In Theorem 3.4,

$$f^{(0)} = \frac{1}{2} \mathbf{F} (\mathbf{R}_0^T \mathbf{V}^T \mathbf{V} \mathbf{R}_0 + \mathbf{R}_1^T \mathbf{V}^T \mathbf{V} \mathbf{R}_1) f$$

and

$$f^{(i)} = f^{(0)} + \frac{1}{2} \mathbf{F} (\mathbf{R}_0^T \mathbf{W}^T \mathbf{W} \mathbf{R}_0 + \mathbf{R}_1^T \mathbf{W}^T \mathbf{W} \mathbf{R}_1) f^{(i-1)}, \forall i \geq 1.$$

If we denote

$$\frac{1}{2} \mathbf{F} (\mathbf{R}_0^T \mathbf{V}^T \mathbf{V} \mathbf{R}_0 + \mathbf{R}_1^T \mathbf{V}^T \mathbf{V} \mathbf{R}_1) = \mathbf{A}$$

and

$$\frac{1}{2} \mathbf{F} (\mathbf{R}_0^T \mathbf{W}^T \mathbf{W} \mathbf{R}_0 + \mathbf{R}_1^T \mathbf{W}^T \mathbf{W} \mathbf{R}_1) = \mathbf{B},$$

then we have

$$f^{(0)} = \mathbf{A}f$$

and

$$f^{(i)} = \mathbf{A}f + \mathbf{B}f^{(i-1)}, \forall i > 1.$$

Consequently,

$$f^{(1)} = \mathbf{A}f + \mathbf{B}f^{(0)} = \mathbf{A}f + \mathbf{B}\mathbf{A}f = (\mathbf{I} + \mathbf{B})\mathbf{A}f,$$

$$f^{(2)} = \mathbf{A}f + \mathbf{B}f^{(1)} = (\mathbf{I} + \mathbf{B} + \mathbf{B}^2)\mathbf{A}f,$$

...

$$f^{(i)} = (\mathbf{I} + \mathbf{B} + \mathbf{B}^2 \dots + \mathbf{B}^i)\mathbf{A}f.$$

Denote

$$\mathbf{S} = \sum_{i=0}^{\infty} \mathbf{B}^i,$$

then we can get

$$f = \mathbf{S}\mathbf{A}f$$

3.3 Matrix Form Algorithm

or

$$f = \mathbf{S}f^{(0)}.$$

As a result, if we are given $f^{(0)}$, and if we can construct \mathbf{S} easily, then the problem is solved. A typical \mathbf{B} matrix of size 8×8 is shown in Eq. 3.2.

$$\mathbf{B} = \begin{pmatrix} 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ -0.25 & 0.50 & -0.25 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & -0.25 & 0.50 & -0.25 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & -0.25 & 0.50 & -0.25 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & -0.25 & 0.50 & -0.25 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 & -0.25 & 0.50 & -0.25 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & -0.25 & 0.50 & -0.25 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \end{pmatrix}. \quad (3.2)$$

The \mathbf{S} matrix for very large n , corresponding to the \mathbf{B} matrix as shown in Eq. 3.2, is

$$\mathbf{S} = \begin{pmatrix} 1.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ -0.8571 & 3.4286 & -2.8571 & 2.2857 & -1.7143 & 1.1429 & -0.5714 & 0.1429 \\ 0.7143 & -2.8571 & 5.7143 & -4.5714 & 3.4286 & -2.2857 & 1.1429 & -0.2857 \\ -0.5714 & 2.2857 & -4.5714 & 6.8571 & -5.1429 & 3.4286 & -1.7143 & 0.4286 \\ 0.4286 & -1.7143 & 3.4286 & -5.1429 & 6.8571 & -4.5714 & 2.2857 & -0.5714 \\ -0.2857 & 1.1429 & -2.2857 & 3.4286 & -4.5714 & 5.7143 & -2.8571 & 0.7143 \\ 0.1429 & -0.5714 & 1.1429 & -1.7143 & 2.2857 & -2.8571 & 3.4286 & -0.8571 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 1.0000 \end{pmatrix}. \quad (3.3)$$

It is hard to see the pattern of the \mathbf{S} matrix as shown in Eq. 3.3, but if all the elements of the \mathbf{S} matrix are multiplied by $n - 1$ ($n = 8$ here), then a new matrix $\tilde{\mathbf{S}}$ is shown in Eq. 3.4.

3.3 Matrix Form Algorithm

$$\tilde{\mathbf{S}} = \begin{pmatrix} 7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -6 & 24 & -20 & 16 & -12 & 8 & -4 & 1 \\ 5 & -20 & 40 & -32 & 24 & -16 & 8 & -2 \\ -4 & 16 & -32 & 48 & -36 & 24 & -12 & 3 \\ 3 & -12 & 24 & -36 & 48 & -32 & 16 & -4 \\ -2 & 8 & -16 & 24 & -32 & 40 & -20 & 5 \\ 1 & -4 & 8 & -12 & 16 & -20 & 24 & -6 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 7 \end{pmatrix}. \quad (3.4)$$

Clearly, $\tilde{\mathbf{S}}$ has a regular pattern and it can be simply constructed using the following MATLAB code:

```
S(:,1) = ((n-1):-1:0)';
S(:,n) = (0:n-1)';
S(1,2:n) = zeros(1,n-1);
S(n,1:n-1) = zeros(1,n-1);

for i = 1:n
    S(i,1) = S(i,1)*(-1)^(i+1);
    S(i,n) = S(i,n)*(-1)^(i+n);
end

for i = 2:n-1
    for j = 2:i
        S(i,j) = 4*(n-i)*(j-1)*(-1)^(i+j);
        S(j,i) = S(i,j);
    end
end
```

This $\tilde{\mathbf{S}}$ matrix constructing algorithm has been numerically tested but without mathematical proof.

For 1D multilevel up super-resolution, it is easy to see that they can be processed

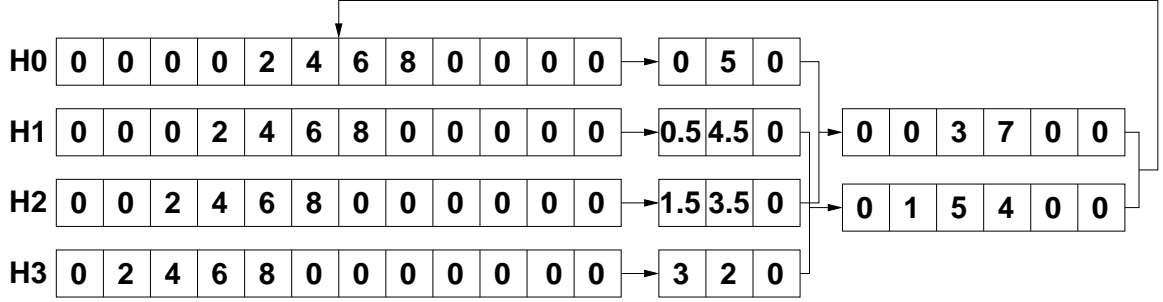


Figure 14: Decomposition of 1D level-2 up super-resolution problem into two level-1 up process.

as multiple level-1 up super-resolution as shown in Fig. 14. In Fig. 14, the original 1D image “H0” is shifted left three time to prepare the four level-2 down low resolution images which have only pixels as shown in the middle. According to their relative shifting position, these low resolution images can be combined pairwise to get the level-1 up super-resolution image, and this process can be continued until the original high resolution image “H0” is obtained.

Using this one step solution, 1D image super-resolution can be solved easily providing the relative positions are known for all the low resolution images.

3.3.6 2D and Level-1 Up Super-Resolution

Since the 1D matrix form algorithm is equivalent to the intuitive algorithm and the intuitive idea is a general idea which can be applied easily in the 2D case, the matrix form of the 2D level-1 super-resolution algorithm can be derived similar to the steps in Section 3.3.2. Here we skip all the detailed steps and just show the results.

Theorem 3.10 *Let $\mathbf{M} \in \mathfrak{R}^{n \times n}$ be the original square image, $\mathbf{R}_0 \in \mathfrak{R}^{n \times n}$ and $\mathbf{R}_1 \in \mathfrak{R}^{n \times n}$ are two shift matrices. $\mathbf{L}_{ij} \in \mathfrak{R}^{\frac{n}{2} \times \frac{n}{2}}$ is the ij th low resolution image. $\mathbf{V} \in \mathfrak{R}^{\frac{n}{2} \times n}$ and $\mathbf{W} \in \mathfrak{R}^{\frac{n}{2} \times n}$ are the scaling and wavelet matrices. Let $\mathbf{P}_i = \mathbf{V}\mathbf{R}_i$, $\mathbf{Q}_i = \mathbf{V}\mathbf{R}_i^T$,*

3.3 Matrix Form Algorithm

$\mathbf{S}_i = \mathbf{W}\mathbf{R}_i$, $\mathbf{T}_i = \mathbf{W}\mathbf{R}_i^T$; and further, $\mathfrak{A} = \mathbf{P}_0^T \mathbf{P}_0 + \mathbf{P}_1^T \mathbf{P}_1$, $\mathfrak{B} = \mathbf{S}_0^T \mathbf{S}_0 + \mathbf{S}_1^T \mathbf{S}_1$, $\mathfrak{C} = \mathbf{Q}_0^T \mathbf{Q}_0 + \mathbf{Q}_1^T \mathbf{Q}_1$, $\mathfrak{D} = \mathbf{T}_0^T \mathbf{T}_0 + \mathbf{T}_1^T \mathbf{T}_1$. Then,

1. $\mathbf{L}_{ij} = \mathbf{P}_i \mathbf{M} \mathbf{Q}_j^T$, $i, j = 0, 1$
2. $\mathbf{M}^{(0)} = \frac{1}{4} \mathbf{F} \mathfrak{A} \mathbf{M} \mathfrak{C} = \frac{1}{4} \mathbf{F} \sum_{i,j=0}^1 \mathbf{P}_i^T \mathbf{L}_{ij} \mathbf{Q}_j$
3. $\mathbf{M}^{(s)} = \mathbf{M}^{(0)} + \frac{1}{4} \mathbf{F} [\mathfrak{B} \mathbf{M}^{(s-1)} \mathfrak{C} + (\mathfrak{A} + \mathfrak{B}) \mathbf{M}^{(s-1)} \mathfrak{D}]$
4. $\mathbf{M} = \lim_{s \rightarrow \infty} \mathbf{M}^{(s)}$

3.3.7 2D and Multilevel Up Super-Resolution

Similar to the 2D level-1 up super-resolution algorithm, the 2D multilevel up super-resolution algorithm can be derived in the same way, and only the results are shown in the following theorem.

Theorem 3.11 Let $\mathbf{M} \in \mathfrak{R}^{n \times n}$ be the original square image and define $\mathbf{A}_1 = \mathbf{V}_1$, $\mathbf{A}_2 = \mathbf{V}_2 \mathbf{A}_1$, \dots , $\mathbf{A}_k = \mathbf{V}_k \mathbf{A}_{k-1}$, similarly, $\mathbf{B}_1 = \mathbf{W}_1$, $\mathbf{B}_2 = \mathbf{W}_2 \mathbf{A}_1$, \dots , $\mathbf{B}_k = \mathbf{W}_k \mathbf{A}_{k-1}$ where k is the down sampling level. Also define $\mathbf{P}_{li} = \mathbf{A}_l \mathbf{R}_i$, $\mathbf{Q}_{li} = \mathbf{A}_l \mathbf{R}_i^T$, $\mathbf{S}_{li} = \mathbf{B}_l \mathbf{R}_i$, $\mathbf{T}_{li} = \mathbf{B}_l \mathbf{R}_i^T$. Further, $\mathfrak{A}_l = \sum_{i=0}^{2^k-1} \mathbf{P}_{li}^T \mathbf{P}_{li}$, $\mathfrak{B}_l = \sum_{i=0}^{2^k-1} \mathbf{S}_{li}^T \mathbf{S}_{li}$, $\mathfrak{C}_l = \sum_{i=0}^{2^k-1} \mathbf{Q}_{li}^T \mathbf{Q}_{li}$, $\mathfrak{D}_l = \sum_{i=0}^{2^k-1} \mathbf{T}_{li}^T \mathbf{T}_{li}$. Then,

1. $\mathbf{L}_{ij} = \mathbf{P}_{ki} \mathbf{M} \mathbf{Q}_{kj}^T$, $i, j = 0, 1, \dots, 2^k - 1$
2. $\mathbf{M}^{(0)} = \frac{1}{4^k} \mathbf{F} \mathfrak{A}_k \mathbf{M} \mathfrak{C}_k = \frac{1}{4^k} \mathbf{F} \sum_{i,j=0}^{2^k-1} \mathbf{P}_{ki}^T \mathbf{L}_{ij} \mathbf{Q}_{kj}$
3. $\mathbf{M}^{(s)} = \mathbf{M}^{(0)} + \frac{1}{4^k} \mathbf{F} \sum_{l=1}^k [\mathfrak{B}_l \mathbf{M}^{(s-1)} \mathfrak{C}_l + (\mathfrak{A}_l + \mathfrak{B}_l) \mathbf{M}^{(s-1)} \mathfrak{D}_l]$
4. $\mathbf{M} = \lim_{s \rightarrow \infty} \mathbf{M}^{(s)}$

4 Summary

In this thesis, a one step algorithm to solve the 1D image (without blurring) super-resolution problem with the assumption that the relative positions of all the low resolution images are known. This algorithm only needs the usual matrix vector products and no other complicated operations. It is a one step solution and the solution should be the exact original image. This simple algorithm is based on an intuitive idea to perform image super-resolution. By developing the intuitive algorithm to the matrix form algorithm, and analyzing the final form, the algorithm is proposed without mathematical proof. In the 2D image super-resolution case, only a matrix form iteration algorithm is provided for both level-1 and multilevel up image super-resolution. The algorithm converges slowly due to the intrinsic similarity between the algorithm and the *Landweber Iteration*. Though no one step algorithm is proposed for the 2D image super-resolution, the 1D algorithm may shine some light on the 2D case. For example, the \mathbf{S} matrix in the 1D case may be used as a preconditioner for the Conjugate-Gradient method. In addition, if it is possible to convert the 2D image super-resolution problem to the 1D case, then the one step algorithm can be applied easily to it and finally solve the problem.

References

- [Baker 1977] C.T.H. Baker. *The numerical treatment of integral equations*. Claredon Press, Oxford, UK, 1977.
- [Berman 1994] D. Berman, J. Bartell, and D. Salesin. Multiresolution painting and compositing. In *Proceedings of SIGGRAPH*, volume 94, pages 85–90, New York, 1994.
- [Bose 1993] N. K. Bose, H. C. Kim, and H. M. Valenzuela. Recursive Total Least Squares Algorithm for Image Reconstruction from Noisy, Undersampled Multiframe. *Multidimensional Systems and Signal Processing* **4**(July 1993), 253–268.
- [Bose 2004] N.K. Bose and M.B. Chappalli. A second-generation wavelet framework for super-resolution with noise filtering. *Int. J. Imaging Syst. Technol.* **14**(2004), 84–89.
- [Brown 1981] J. L. Brown. Multichannel sampling of low-pass signals. *IEEE Trans. CAS* **28**(1981), 101–106.
- [Chang 2006] S. G. Chang, Z. Cvetkovic, and M. Vetterli. Locally adaptive wavelet-based image interpolation. *IEEE Trans. Image Process.* **15**(2006), 1471–1485.
- [Cheeseman 1996] P. Cheeseman, B. Kanefsky, R. Kraft, J. Stutz, and R. Hanson. Super-resolved surface reconstruction from multiple images. In *Maximum Entropy and Bayesian*

REFERENCES

- Methods*, pages 298–308, Santa Barbara, CA, 1996. Kluwer.
- [Christensen 1995] P. H. Christensen, E. J. Stollnitz, D. H. Salesin, and T. D. DeRose. Wavelet radiance. In G. Sakas, P. Shirley, and S. Müller, editors, *Photorealistic Rendering Techniques*, pages 295–309, Berlin, 1995. Springer-Verlag.
- [Chung 2006] J. Chung, E. Haber, and J. Nagy. Numerical methods for coupled super-resolution. *Inverse Problem* **22**(2006), 1261–1272.
- [Daubechies 1988] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics* **41**(October 1988), 909–996.
- [DeVore 1992] R. DeVore, B. Jawerth, and B. Lucier. Image compression through wavelet transform coding. *IEEE Transactions on Information Theory* **38**(March 1992), 719–746.
- [El-Khamy 2005] S.E. El-Khamy, M.M. Hadhoud, M.I. Dessouky, B.M. Salam, and F.E. Abd El-Samie. Blind multichannel reconstruction of high-resolution images using wavelet fusion. *Appl. Optics* **44**(2005), 7349–7356.
- [Elad 1997] M. Elad and A. Feuer. Restoration of a Single Superresolution Image from Several Blurred, Noisy, and Undersampled Measured Images. *IEEE Trans. IP* **6**(1997), 1646–1658.

REFERENCES

- [Eren 1997] P. E. Eren, M. I. Sezan, and A. Tekalp. Robust. Object Based High-Resolution Image Reconstruction from LowResolution Video. *IEEE Trans. IP* **6**(1997), 1446–1451.
- [Finkelstein 1994] A. Finkelstein and D. H. Salesin. Multiresolution curves. In *Proceedings of SIGGRAPH*, volume 94, pages 261–268, New York, 1994. ACM.
- [Goldstein 1964] A. A. Goldstein. Convex programming in Hilbert space. *Bull. Amer. Math. Soc.* **70**(1964), 709–710.
- [Gortler and Cohen 1995] S. J. Gortler and M. F. Cohen. Hierarchical and variational geometric modeling with wavelets. In *Proceedings of the 1995 Symposium on Interactive 3D Graphics*, pages 35–42, New York, 1995. ACM.
- [Gortler 1995] S. J. Gortler, P. Schröder, M. F. Cohen, and P. Hanrahan. Wavelet radiosity. In *Proceedings of SIGGRAPH*, volume 93, pages 221–230, New York, 1995. ACM.
- [Hardie 1997] R. C. Hardie, K. J. Barnard, and E. E. Armstrong. Joint MAP Registration and High-Resolution Image Estimation Using a Sequence of Undersampled Images. *IEEE Trans. IP* **6**(Dec. 1997), 1621–1633.
- [Irani and Peleg 1993] M. Irani and S. Peleg. Motion analysis for image enhancement: Resolution, occlusion and transparency. *Journal of*

REFERENCES

- Visual Communications and Image Representation* **4**(Dec. 1993), 324–335.
- [Jacobs 1995] C. E. Jacobs, A. Finkelstein, and D. H. Salesin. Fast multiresolution image querying. In *Proceedings of SIGGRAPH*, volume 95, pages 277–286, New York, 1995. ACM.
- [Kim 1990] S. P. Kim, N. K. Bose, and H. M. Valenzuela. Recursive reconstruction of high resolution image from noisy under-sampled multiframe. *IEEE Trans. ASSP* **38**(1990), 1013–1027.
- [Kim and Su 1993] S. P. Kim and W.-Y. Su. Recursive high-resolution reconstruction of blurred multiframe images. *IEEE Trans. IP* **2**(1993), 534–539.
- [Komatsu 1993] T. Komatsu, K. Aizawa, T. Igarashi, and T. Saito. Signal-processing based method for acquiring very high resolution image with multiple cameras and its theoretical analysis. In *Proc. Inst. Elec. Eng.*, volume 140, pages 19–25, 1993. pt. I.
- [Liu 1994] Z. Liu, S. J. Gortler, and M. F. Cohen. Hierarchical space-time control. In *Proceedings of SIGGRAPH*, volume 94, pages 35–42, New York, 1994. ACM.

REFERENCES

- [Lukosz 1966] W. Lukosz. Optical systems with resolving power exceeding the classical limit. *J. Opt. Soc. Am.* **56**(1966), 1463–1472.
- [Lukosz 1967] W. Lukosz. Optical systems with resolving power exceeding the classical limit II. *J. Opt. Soc. Am.* **57**(1967), 932–941.
- [Mallet 1989] S. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **11**(July 1989), 673–693.
- [Meyers 1994] D. Meyers. Multiresolution tiling. *Computer Graphics Forum* **13**(December 1994), 325–340.
- [Nguyen 2000] N. Nguyen and P. Milanfar. A wavelet-based interpolation-restoration method for superresolution (wavelet superresolution). *Circuits Syst. Signal Process.* **19**(2000), 321–338.
- [Park 2003] S. C. Park, M. K. Park, and M. G. Kang. Super-resolution image reconstruction: a technical overview. *Signal Processing Magazine, IEEE* **20**(May 2003), 21–36. and reference therein.
- [Patti 1997] A. J. Patti, M. I. Sezan, and A. M. Tekalp. Superresolution Video Reconstruction with Arbitrary Sampling Lat-

REFERENCES

- tices and Nonzero Aperture Time. *IEEE Trans. IP* **6**(Aug. 1997), 1064–1076.
- [Patti 1998] A. J. Patti, A. M. Tekalp, and M. I. Sezan. A New Motion Compensated Reduced Order Model Kalman Filter for Space-Varying Restoration of Progressive and Interlaced Video. *IEEE Trans. IP* **7**(Apr. 1998), 543–554.
- [Schröder 1994] P. Schröder, S. J. Gortler, M. F. Cohen, and Pat Hanrahan. Wavelet projections for radiosity. *Computer Graphics Forum* **13**(June 1994), 141–151.
- [Schultz 1996] R. R. Schultz and R. L. Stevenson. Extraction of high-resolution frames from video sequences. *IEEE Trans. IP* **5**(June 1996), 996–1011.
- [Shen 2004] L.X. Shen and Q.Y. Sun. Biorthogonal wavelet system for high-resolution image reconstruction. *IEEE Trans. Signal Process.* **52**(2004), 1997–2011.
- [Tekalp 1992] A. M. Tekalp, M. K. Ozkan, and M. I. Sezan. High-resolution image reconstruction from lower-resolution image sequences and space-varying image restoration. In *IEEE Int. Conf. Acoustics*, volume III of *Speech and Signal Processing (ICASSP)*, pages 169–172, San Francisco, CA, Mar 1992.

REFERENCES

- [Tom 1994] B. C. Tom and A. K. Katsaggelos. Reconstruction of a high resolution image from multiple degraded mis-registered low resolution images. In *SPIE VCIP*, volume 2308, pages 971–981, Chicago, Sept. 1994.
- [Tom 1996] B. C. Tom and A. K. Katsaggelos. An Iterative Algorithm for Improving the Resolution of Video Sequences. In *SPIE VCIP*, volume 2727, pages 1430–1438, Orlando, FL, Mar. 1996.
- [Tsai and Huang 1984] R.Y. Tsai and T.S. Huang. Multiframe image restoration and registration. In R.Y. Tsai and T.S. Huang, editors, *Advances in Computer Vision and Image Processing*, volume 1, pages 317–339. JAI Press Inc., 1984.
- [Ur and Gross 1992] H. Ur and D. Gross. Improved resolution from subpixel shifted pictures. In *CVGIP: Graphical Models and Image Processing*, volume 54, pages 181–186, Orlando, FL, Mar. 1992. Academic Press, Inc.
- [Watanabe 2006] K. Watanabe, Y. Iwai, H. Nagahara, M. Yachida, and T. Suzuki. Video synthesis with high spatio-temporal resolution using motion compensation and image fusion in wavelet domain. *LECT NOTE COMPUT SCI* **3851**(2006), 480–489.